"UNIVERSIDAD NACIONAL DE PIURA"

FACULTAD DE CIENCIAS

ESCUELA PROFESIONAL DE INGENIERÍA ELECTRÓNICA Y TELECOMUNICACIONES

"Diseño de un sistema de seguridad ciudadana usando las Tecnologías de la Información para la prevención de delitos contra las personas y bienes, con participación ciudadana.".

PRESENTADA POR:

BACH. JOSE ANGEL PAZ DURAND

BACH. MIGUEL ANGEL AYALA PALOMINO

TESIS PARA OBTAR EL TÍTULO PROFESIONAL DE INGENIERO ELECTRÓNICO Y TELECOMUNICACIONES

PIURA – PERÚ

2015

TESIS PRESENTADA COMO REQUISITO PARA OPTAR EL TÍTULO DE INGENIERO ELECTRÓNICO Y TELECOMUNICACIONES

ASESOR:

ING. FRANKLIN BARRA ZAPATA

TESISTA:

BACH. JOSE ANGEL PAZ DURAND

BACH. MIGUEL ANGEL AYALA PALOMINO

JURADO EVALUADOR:

PRESIDENTE:

ING. MIGUEL PANDURO ALVARADO

SECRETARIO:

ING. CESAR HUMBERTO ESTRADA CRISANTO

VOCAL:

ING. AYAX MANUEL SIFUENTES MONTES

DEDICATORIA

A mis padres, tías y abuelitos por su esfuerzo y apoyo incondicional a lo largo de mi carrera.

José Ángel.

DEDICATORIA

Especialmente dedicado al Sr. Angel Ayala, mi padre; por su apoyo incondicional, sus consejos y por todo el amor y cariño brindado así también a ,mi querida madre, hermanos, a mis pequeñas Angie Valentina y Cielo Maribel, a mi esposa; y a todos mis amigos por su apoyo y cariño y todos los momentos compartidos.

Miguel Ángel

AGRADECIMIENTO

En primer lugar a Dios por darnos la vida y dicha de compartir día a día con quienes más queremos y por la oportunidad de ser mejores cada dia, de llevar a cabo nuestras metas y el sueño de ser un profesional

Esta tesis no hubiera sido posible sin el impulso de todas aquellas personas que nos han ayudado personalmente en el desarrollo del trabajo de esta índole, comenzando por el Ing. Franklin Barra Zapata a quien agradecemos su apoyo en la elaboración del presente sistema de monitoreo.

Un agradecimiento muy especial por la paciencia y el ánimo recibidos de nuestra familia y amigos.

A todos ellos, muchas gracias.

INTRODUCCIÓN

La criminalidad y violencia en el mundo constituyen en la actualidad un problema político social de primer orden, que exige la necesidad de implementar medidas Concretas para disminuir la violencia urbana en las principales ciudades del país, en particular contra la delincuencia común, cuyos efectos los padece transversalmente toda la población.

Esta violencia obedece a muchos factores causales de índole socioeconómico y cultural, donde la familia, la escuela, la comunidad y los medios de comunicación constituyen espacios de socialización muy importantes; sin embargo, éstos históricamente no han articulado una clara orientación de sus objetivos, contribuyendo a una débil formación ciudadana.

La criminalidad y la delincuencia urbana es una de las manifestaciones más notorias de la violencia contemporánea. Las ciudades enfrentan altas tasas de delincuencia que amenazan los sentimientos de seguridad de la población. Vernos libres de la delincuencia, gozar de un ambiente de tranquilidad, estar protegido contra la violencia en el hogar y en la calle, lograr que las ciudades sean más seguras son ingredientes indispensables para un desarrollo sostenido.

Históricamente las ciudades siempre han sufrido en mayor o menor dimensión los avatares de la violencia, pero hoy en día, por la incidencia de muchos factores estructurales como la desocupación, falta de empleo, las migraciones, la pérdida de valores, etcétera, han elevado sus índices tornándose más agresivas y temerarias.

Por lo que se plantea desarrollar e implementar un sistema de seguridad basados en las Tecnologías de la Información que contribuyan a prevenir y/o evitar los delitos contra las personas, bienes y con la participación ciudadana.

En la presente tesis se plantea el diseño y construcción de una alarma vecinal comunitaria la cual funciona con la participación de toda una comunidad vecinal además se plantea el desarrollo de aplicativos para el uso personal en caso de emergencias y también para denunciar diferentes tipos de actos de violencia o delictivos y otros que contribuyan al orden y seguridad de la ciudad.

INDICE

INDICE FIGURAS

INDICE DE TABLAS

CAPITULO I

PLANTEAMIENTO DE LA INVESTIGACIÓN

1.1. TITULO DEL PROYECTO

"Diseño de un sistema de seguridad ciudadana usando las Tecnologías de la Información para la prevención de delitos contra las personas y bienes, con participación ciudadana.".

1.2. DESCRIPCIÓN DEL PROBLEMA

A diario, a través de los noticieros televisivos, radiales y de la prensa escrita. Vemos, escuchamos y leemos. Fatídicas noticias, que dan cuenta de asesinatos, violaciones, ajustes de cuentas, asaltos a mano armada, ingreso indebido a la propiedad privada, pornografía infantil. La profesión de sicarios y el requerimiento de sus servicios se han incrementado. A través de los noticieros y de manera inconsulta, las noticias nos llegan a raudales.

Definitivamente la seguridad ciudadana se ha convertido en una de las grandes preocupaciones que tiene el pueblo peruano. Debe responder a una Política de Estado, que tenga en cuenta una política de tolerancia cero y una política preventiva. Y el liderazgo debe recaer en el gobierno nacional que además no debe esquivar su responsabilidad. Sin embargo, urge desarrollar e implementar un modelo que nos ayude a garantizar la seguridad de nuestras familias y en especial de los más débiles que son nuestros hijos.

La inseguridad es un fenómeno específico de las realidades urbanas de todo el mundo, es un factor condicionante al desarrollo de una nación, al mejoramiento de la calidad de vida y al crecimiento económico.

Por lo tanto el presente perfil plantea implementar un sistema de seguridad Utilizando las tecnologías de la información y con la participación activa de la ciudadanía.

1.3. ANTECEDENTES

Según el **Plan Nacional de Seguridad de Ciudadana 2013-2018** del Ministerio del Interior nos dice lo siguiente: La seguridad ciudadana ha sido y es una de las principales demandas de la población peruana. Ha ocupado un lugar importante en la agenda gubernamental de los últimos veinte años. A lo largo de este período, ha merecido diversos enfoques y tratamientos por parte de las autoridades, pero con escaso éxito hasta hoy. Y es que la seguridad ciudadana es un fenómeno social complejo, multidimensional y multicausal, que, por ello, debe ser abordado desde diversos ámbitos de forma simultánea.

A la luz de la experiencia de los últimos años, una primera constatación es que la seguridad ciudadana no es solo una política de un determinado gobierno, sino, esencialmente, una política de Estado. No es un problema que merezca solo una solución policial de corto plazo, sino que supone un proceso de mediano y largo plazo, con la complejidad que ello supone. Queda claro que, además del diseño e implementación de una solución bajo un enfoque multidimensional, se requiere el monitoreo y la evaluación permanentes en un proceso de gestión por resultados.

Mantener un mapa delictual actualizado es fundamental para generar enfoques adecuados y eficientes. Según el mapa del delito, en la actualidad, la mayor incidencia, tanto en delitos como en faltas, son contra el patrimonio contra la vida, el cuerpo y la salud de las personas, contra la libertad y contra la seguridad pública.

Como es de conocimiento público, desde hace una década el Perú se encuentra en un proceso sostenido de crecimiento económico. Este se expresa en una creciente inversión nacional y extranjera, así como también en un mayor movimiento de transacciones comerciales y financieras. Existen más oportunidades de trabajo. El

ingreso promedio per cápita se incrementa, y esto se refleja en los hogares y empresas. Lamentablemente, a la par del desarrollo y crecimiento económico, la delincuencia también ha aumentado, con el consiguiente incremento de la inseguridad ciudadana. Los peruanos se sienten más inseguros en sus hogares, centros de trabajo y en los principales espacios públicos.

Según Alejandro Prince en su artículo titulado: **"Las TIC y su relación con la Seguridad Ciudadana: un marco de análisis a la problemática"**

Indica lo siguiente:

Mucho se ha escrito de los efectos beneficiosos de las TIC sobre la seguridad, en el contexto de la Sociedad de la Información o como preferimos, del Conocimiento; así como de la eficiencia y transparencia que las TIC aportarán a la Administración Pública y al Gobierno en todos sus niveles y áreas. Sin embargo, al hablar de la injerencia de las Nuevas Tecnologías en la seguridad pública, debemos tener en cuenta el contexto -tanto político-administrativo como social- en el cual se insertan, con la finalidad de no caer en "soluciones mágicas" ni recetas universales.

La creciente inclusión digital por parte de la ciudadanía, la adopción de Nuevas Tecnologías y sus consecuentes conocimientos, y la vertiginosa rapidez con las cuales estas modifican las relaciones entre las personas, demandan una adecuación y desarrollo evolutivo del Estado en esta temática. El Estado no sólo es un ente jurídico y social que regula y condiciona el accionar de sus ciudadanos sino que también es influido por los procesos sociales, lo que obliga a recomponer sus estructuras y acciones de manera continua frente a las nuevas realidades.

1.4. FORMULACIÓN DEL PROBLEMA

¿Es posible diseñar e implementar un sistema de seguridad ciudadana usando las Tecnologías de la Información para la prevención de delitos contra las personas y bienes y poder contribuir con la seguridad en la ciudad con participación ciudadana?

1.5. HIPÓTESIS

Usando las tecnologías existentes, si es posible diseñar un sistema de seguridad ciudadana usando las Tecnologías de la Información para la prevención de delitos contra las personas y bienes y poder contribuir con la seguridad en la ciudad

1.6. OBJETIVOS

1.6.1. OBJETIVOS GENERALES

Diseñar un sistema de seguridad ciudadana usando las Tecnologías de la Información para la prevención de delitos contra las personas y bienes y poder contribuir con la seguridad en la ciudad

1.6.2. OBJETIVOS ESPECIFICOS

- Diseñar hardware electrónico para alarma vecinal con tecnología GSM y Android.
- Diseñar sistema de comunicación entre el usuario y hardware usando tecnologías de la información como comunicación GSM
- Diseñar software necesario para el control del sistema de seguridad
- Desarrollar aplicativos en plataforma Android que contribuyan con la seguridad ciudadana.

CAPITULO II

MARCO TEÓRICO

2.1 MICROCONTROLADORES

2.1.1 ¿QUÉ ES UN MICROCONTROLADOR?

El microcontrolador es un circuito integrado de muy alta escala de integración que contiene las partes funcionales de un computador:

- CPU (Central Processor Unit o Unidad de Procesamiento Central)
- Memorias volátiles (RAM), para datos
- Memorias no volátiles (ROM, PROM, EPROM) para escribir el programa
- Líneas de entrada y salida para comunicarse con el mundo exterior.
- Algunos periféricos (comunicación serial, temporizador, convertidor A/D, etc.)

"Es decir el microcontrolador es un computador integrado en un solo chip. Integrar todos estos elementos en un solo circuito integrado ha significado desarrollar aplicaciones importantes en la industria al economizar materiales, tiempo y espacio". [1]

2.1.2 ¿QUE NO HACE EL MICROCONTROLADOR?

Las aplicaciones de un microcontrolador son tan inmensas que el límite es la propia imaginación del usuario. Estos microcontroladores están en el auto, en el televisor, en el teléfono, en una impresora, en un horno de microondas, en un juguete, en un transbordador espacial etc.

Los siguientes son algunos campos en los que los microcontroladores tienen gran uso:

- En la industria del automóvil: Control de motor, alarmas, regulador del servofreno, dosificador, etc.
- En la industria de los electrodomésticos: control de calefacciones, lavadoras, cocinas eléctricas, etc.
- En informática: como controlador de periféricos. Controlar impresoras, plotters, cámaras, scanners, terminales, unidades de disco, teclados, módems, etc.

- En la industria de imagen y sonido: tratamiento de la imagen y sonido, control de los motores de arrastre del giradiscos, magnetófono, video, etc.

En la industria, en general se utilizan en:

- Regulación: todas las familias de microcontroladores incorporan en alguna de sus versiones conversores A/D y D/A, para la regulación de la velocidad de las máquinas, de niveles, de temperatura, etc.
- Automatismos: La enorme cantidad de líneas de entrada y salidas, y su inmunidad al ruido le hacen muy valioso para el control secuencial de procesos. Por ejemplo control de máquinas, herramientas, apertura y cierre automático de puertas según condiciones, plantas empaquetadoras, aparatos de maniobra de ascensores, etc.
- Robótica: para control de los motores y captura de señales de los diferentes sensores, fabricación de controladores robóticos para sistemas automáticos, etc.

Instrumentos portátiles compactos:

- Radio paginador numérico (beeper)
- Planímetro electrónico
- Nivelímetro digital
- Identificador-probador de circuitos integrados
- Tacómetro digital
- Panel frontal de un osciloscopio
- Controlador de display LCD
- Analizador de espectros, etc.

Dispositivos autónomos:

- Fotocopiadoras
- Máquinas de escribir
- Selector, Codificador decodificador de TV
- Localizador de peces
- Teléfonos de tarjeta
- Teléfonos celulares
- Cerraduras electrónicas
- Sistemas de seguridad

Se emplea también en medicina, en aplicaciones militares, edificios inteligentes, etc.

2.1.3 PRINCIPALES FABRICANTES

Por lo general los fabricantes de microprocesadores, lo son de microcontroladores. Los fabricantes de microcontroladores son más de 50, podemos mencionar a:

- ATMEL
- MOTOROLA
- INTEL
- MICROCHIP
- NEC
- HITACHI
- MITSIBISHI
- PHILIPS
- MATSUSHITA
- TOSHIBA
- AT&T
- ZILOG
- SIEMENS
- NATIONAL SEMICONDUCTOR, ETC.

2.1.4 APARICIÓN Y DESARROLLO DE LOS MICROCONTROLADORES

2.1.4.1 BREVE ESBOZO HISTÓRICO

La siguiente es una lista cronológica de los eventos tecnológicos más recientes que han tenido impacto sobre la aparición y el desarrollo del campo de los microcontroladores en la electrónica digital.

1971: Intel fabrica el primer microprocesador (el 4004) de tecnología PMOS. Este era un microprocesador de 4 bits y fue fabricado por Intel a petición de Datapoint Corporation con el objeto de sustituir la CPU de terminales inteligentes fabricadas en esa fecha por Datapoint mediante circuitería discreta. El dispositivo fabricado por Intel resultó 10 veces más lento de lo requerido y Datapoint no lo compró, de esta manera Intel comenzó a comercializarlo. El 4004 podía direccionar sólo 4096 (4k) posiciones de memoria de 4 bits, reconocía 45 instrucciones y podía ejecutar una instrucción en 20 µseg en promedio.

1972: Las aplicaciones del 4004 estaban muy limitadas por su reducida capacidad y rápidamente Intel desarrolló una versión más poderosa (el 8008), el cual podía

manipular bytes completos, por lo cual fue un microprocesador de 8 bits. La memoria que este podía manejar se incrementó a 16 kbytes, sin embargo, la velocidad de operación continuó igual.

1973: Intel lanza al mercado el 8080 el primer microprocesador de tecnología NMOS, lo cual permite superar la velocidad de su predecesor (el 8008) por un factor de diez, es decir, el 8080 puede realizar 500K de operaciones por segundo, además se incrementó la capacidad de direccionamiento de memoria a 64 kbytes. A partir del 8080 de Intel se produjo una revolución en el diseño de microcomputadoras y varias compañías fabricantes de circuitos integrados comenzaron a producir microprocesadores. Algunos ejemplos de los primeros microprocesadores son: el IMP-4 y el SC/MP de National Semiconductors, el PPS-4 y PPS-8 de Rockwell International, el MC6800 de Motorola, el F-8 de Fairchild.

1975: Zilog lanza al mercado el Z80, uno de los microprocesadores de 8 bits más poderosos. En ese mismo año, Motorola reduce sus costos con sus microprocesadores 6501 y 6502 (este último adoptado por APPLE para su primera microcomputadora personal). Estos microprocesadores se comercializan en $20 y $25 (DLS USA) respectivamente.

Esto provoca un auge en el mercado de microcomputadoras de uso doméstico y un caos en la proliferación de lenguajes, sistemas operativos y programas (ningún producto era compatible con el de otro fabricante).

1976: Surgen las primeras microcomputadoras de un solo chip, que más tarde se denominarán microcontroladores. Dos de los primeros microcontroladores, son el 8048 de Intel y el 6805R2 de Motorola.

198x: En la década de los 80's comienza la ruptura entre la evolución tecnológica de los microprocesadores y la de los microcontroladores, Ya que los primeros han ido incorporando cada vez más y mejores capacidades para las aplicaciones en donde se requiere el manejo de grandes volúmenes de información y por otro lado, los segundos han incorporado más capacidades que les permiten la interacción con el mundo físico en tiempo real, además de mejores desempeños en ambientes de tipo industrial.

2.1.5 ARQUITECTURA DE UN MICROCONTROLADOR

Según la arquitectura interna de la memoria de un microcontrolador se puede clasificar considerando como el CPU accede a los datos e instrucciones, en 2 tipos: [3]

2.1.5.1 ARQUITECTURA VON NEUMANN

Fue desarrollada por Jon Von Neumann, se caracteriza por tener una sola memoria principal donde se almacenan datos e instrucciones de forma indistinta. La CPU se conecta a través de un sistema de buses (direcciones, datos y control). Esta arquitectura es limitada cuando se demanda rapidez.

Memoria

Bus de direcciones Instrucciones

CPU +

Datos

Bus de datos

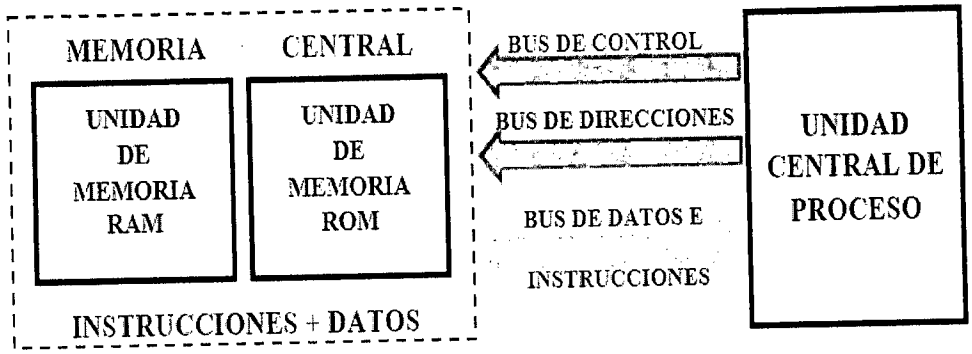


FIG. 2.0 ARQUITECTURA VON NEUMANN

2.1.5.2 ARQUITECTURA HARVARD

Fue desarrollado en Harvard, por Howard Aiken, esta arquitectura se caracteriza por tener 2 memorias independientes una que contiene sólo instrucciones y otra, que contiene sólo datos. Ambas, disponen de sus respectivos sistemas de buses para el acceso y es posible realizar operaciones de acceso simultáneamente en ambas memorias.

Existe una variante de esta arquitectura que permite el acceso a la tabla de datos desde la memoria de programas, es la Arquitectura de Harvard Modificada.

Esta última arquitectura es la dominante en los microcontroladores actuales ya que la memoria de programas es usualmente ROM, OTP, EPROM o FLASH, mientras que la memoria de datos es usualmente RAM.

Por ejemplo las tablas de datos pueden estar en la memoria de programa sin que sean perdidas cada vez que el sistema es apagado.

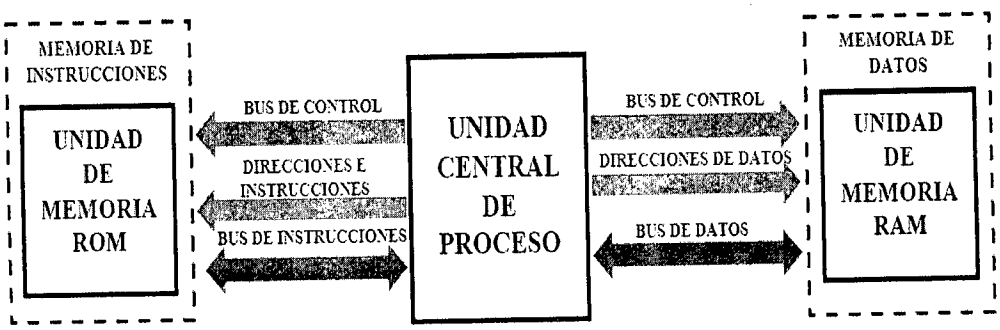


FIG. 2.1 ARQUITECTURA HARVARD

Se puede observar claramente que las principales ventajas de esta arquitectura son:

a) El tamaño de las instrucciones no está relacionado con el de los datos, y por lo tanto puede ser optimizado para que cualquier instrucción ocupe una sola posición de memoria de programa, logrando así mayor velocidad y menor longitud de programa.

b) El tiempo de acceso a las instrucciones puede superponerse con el de los datos, logrando una mayor velocidad de operación.

2.1.6 MICROCONTROLADOR A ELEGIR

2.1.6.1 ¿QUÉ MICROCONTROLADOR ELEGIR?

Sin duda la elección del microcontrolador dependerá de la tarea o proyecto que se tiene en mente pues los fabricantes como se mencionó anteriormente son más de 50, estos tienen muchos modelos enfocados a tareas específicas. Esta selección deberá ir de la mano con factores económicos óptimos así como de la idea del controlador incrustado (embedded controller), el cual es un controlador dedicado a una sola tarea e incorporado al sistema que gobierna.

PROCESAMIENTO DE DATOS: Cuando se desea realizar cálculos complejos en un tiempo limitado, se debe seleccionar un microcontrolador suficientemente rápido para ello. Por otro lado, habrá que tener en cuenta la precisión de los datos a manejar: si

no es suficiente con un microcontrolador de 8 bits, puede ser necesario acudir a microcontroladores de 16 ó 32 bits, o incluso a hardware de coma flotante.

ENTRADA/SALIDA: Se debe identificar la cantidad y tipo de señales a controlar. Una vez realizado este análisis puede ser necesario añadir periféricos externos o cambiar a otro microcontrolador más adecuado a ese sistema.

CONSUMO: algunos productos que incorporan microcontroladores están alimentados con baterías, puede ser que el microcontrolador esté trabajando en estado de bajo consumo pero debe "despertar" ante la activación de una señal (por ejemplo una interrupción) y ejecutar el programa adecuadamente.

MEMORIA: para detectar las necesidades de memoria de una aplicación debemos saber la cantidad y el tipo de memoria necesaria, para esto se debe tener una versión preliminar (pseudo-código) de la aplicación y escoger el microcontrolador apropiado.

ANCHO DE PALABRA: el criterio de diseño debe ser seleccionar el microcontrolador de menor ancho de palabra que satisfaga los requerimientos de la aplicación. Usar un microcontrolador de 4 bits supondrá reducir los costos, mientras que uno de 8 bits puede ser el más adecuado si el ancho de los datos es de un byte. Los microcontroladores de 16 y 32 bits, debido a su elevado costo, deben reservarse para aplicaciones que requieran altas prestaciones (Entrada/Salida grande o espacio de direccionamiento muy elevado).

DISEÑO DE LA PLACA: la selección de un microcontrolador concreto condicionará el diseño de la placa de circuitos impresos.

2.1.7 EL MICROCONTROLADOR 16F877A

El microcontrolador PIC16F877 de Microchip pertenece a una gran familia de microcontroladores de 8 bits (bus de datos) que tienen las siguientes características generales que los distinguen de otras familias:

- Arquitectura Harvard
- Tecnología RISC
- Tecnología CMOS

Estas características se conjugan para lograr un dispositivo altamente eficiente en el uso de la memoria de datos y programa y por lo tanto en la velocidad de ejecución.

El PIC16F877 es un microcontrolador con memoria de programa tipo FLASH, lo que representa gran facilidad en el desarrollo de prototipos y en su aprendizaje ya que no se requiere borrarlo con luz ultravioleta como las versiones EPROM, sino que permite reprogramarlo nuevamente sin ser borrado con anterioridad.

"El PIC16F877 es un microcontrolador de tecnología Microchip fabricado en tecnología CMOS, su consumo de potencia es muy bajo y además es completamente estático, esto quiere decir que el reloj puede detenerse y los datos de la memoria no se pierden".[4]

El encapsulado más común para este microcontrolador es el DIP (Dual In-line Pin) de 40 pines, propio para usarlo en experimentación. La referencia completa es PIC16F877-04 para el dispositivo que utiliza cristal oscilador de hasta 4 MHz, PIC16F877-20 para el dispositivo que utiliza cristal oscilador de hasta 20 MHz o PIC16F877A-I para el dispositivo tipo industrial que puede trabajar hasta a 20 MHz.

Sin embargo, hay otros tipos de encapsulado que se pueden utilizar según el diseño y la aplicación que se quiere realizar. Por ejemplo, el encapsulado tipo surface mount (montaje superficial) tiene un reducido tamaño y bajo costo, que lo hace propio para producciones en serie o para utilizarlo en lugares de espacio muy reducido.

2.1.7.1 CARACTERÍSTICAS PRINCIPALES

- CPU RISC de alta performance
- Set de 35 instrucciones
- Todas las instrucciones son de un ciclo salvo aquellas que incluyen saltos que son de 2 ciclos.
- Velocidad de Trabajo:
- DC - 20 MHz de reloj de entrada
- DC - 200 ns ciclo de instrucción
- Hasta 8K x 14 de trabajo y memoria FLASH de programación.
- Hasta 368 x 8 bytes, data de memoria (RAM)
- Hasta 256 x 8 bytes de Datos en Memoria EEPROM, manejo de Interrupciones (hasta 14 fuentes)
- Unidades aisladoras en hardware de 8 niveles
- Modo de direccionamiento directo, indirecto y relativo.
- Power-on Reset (POR)

- Power-up Timer (PWRT) y Oscilador Start- up Timer (OST)
- Watchdog Timer (WDT) con el reloj RC interno para mejor seguridad.
- Protección de código programable.
- Programación serial vía 2 pines, programación serial en el circuito (ICSP)
- In-Circuit Debugging vía 2 pines
- Amplio rango de voltaje de trabajo: 2.0V a 5.5V

2.1.7.2 CARACTERÍSTICAS PERIFÉRICAS

- Timer0: 8-bit timer/counter con 8-bit de pre-escala
- Timer1: 16-bit timer/counter con pre-escala, que puede ser incrementado durante el modo SLEEP vía reloj externo.
- Timer2: 8-bit timer/counter con registro de período de 8-bit, prescaler y postscaler
- Dos módulos de captura, comparativa, PWM
- Captura de 16-bit, Máx. Resolución: 12.5 ns
- Compara 16-bit, máx. Resolución: 200 ns
- PWM máx. Resolución: 10-bit
- Convertidor Analógico - Digital de 10-bit multicanal
- Puerto de sincronización serial (SSP) con SPI (Modo maestro) e I2C (Maestro/Esclavo)
- Universal Synchronous Asynchronous Receiver Transmitter (USART/SCI) con detección de direcciones de 9-bit
- Puerto paralelo esclavo (PSP) de 8-bits de ancho, con controles externos de RD, WR y CS (solo 40/44-pin)
- Brown-out detection circuitry para Brown-out Reset (BOR)

2.1.7.3 CONFIGURACIÓN DE PINES

Los pines de entrada/salida de este microcontrolador están organizados en cinco puertos, el puerto A con 6 líneas, el puerto B con 8 líneas, el puerto C con 8 líneas, el puerto D con 8 líneas y el puerto E con 3 líneas.

Cada pin de esos puertos se puede configurar como entrada o como salida independiente programando un par de registros diseñados para tal fin. En ese registro un bit en "0" configura el pin del puerto correspondiente como salida y un bit en "1" lo configura como entrada. Dichos pines del microcontrolador también pueden cumplir otras funciones especiales, siempre y cuando se configuren para ello, según se verá más adelante.

Los pines del puerto A y del puerto E pueden trabajar como entradas para el convertidor Análogo a Digital interno, es decir, allí se podría conectar una señal proveniente de un sensor o de un circuito analógico para que el microcontrolador la convierta en su equivalente digital y pueda realizar algún proceso de control o de instrumentación digital. El pin RB0/INT se puede configurar por software para que funcione como interrupción externa, para configurarlo se utilizan unos bits de los registros que controlan las interrupciones.

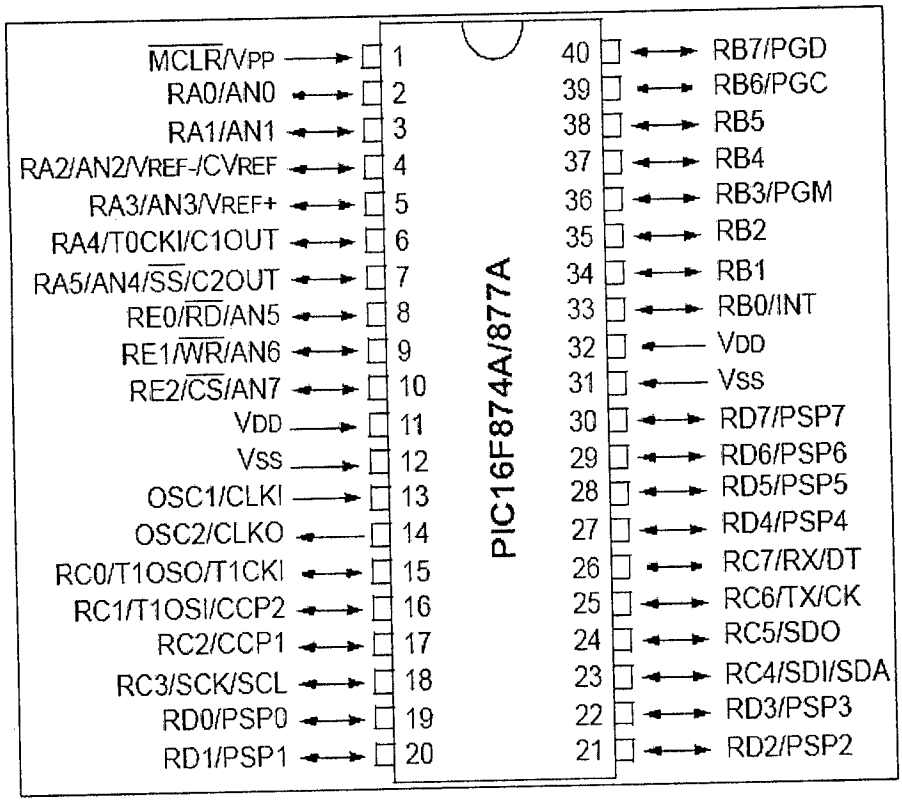


FIG. 2.2 DISTRIBUCIÓN DE PINES DEL PIC16F877A

El pin RA4/TOCKI del puerto A puede ser configurado como un pin de entrada/salida o como entrada del temporizador/contador. Cuando este pin se programa como entrada digital, funciona como un disparador de Schmitt (Schmitt trigger), puede reconocer señales un poco distorsionadas y llevarlas a niveles lógicos (cero y cinco voltios).

Cuando se usa como salida digital se comporta como colector abierto (open collector), por lo tanto, se debe poner una resistencia de pull-up (resistencia externa conectada a un nivel de cinco voltios).

Como salida, la lógica es inversa: un "0" escrito al pin del puerto entrega en el pin un

"1" lógico. Además, como salida no puede manejar cargas como fuente, sólo en el modo sumidero.

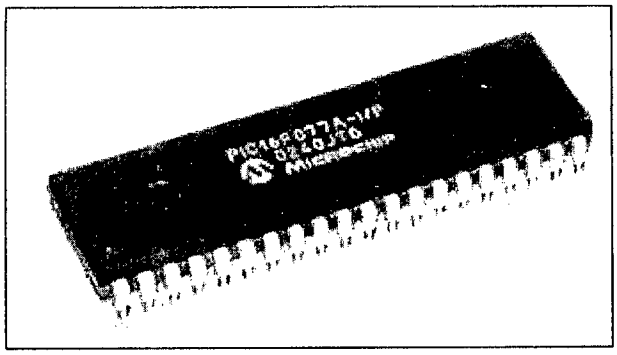


FIG. 2.3 FOTOGRAFIA DEL PIC 16F877A

	PUERTO A	PUERTO B	PUERTO C	PUERTO D
Modo sumidero	150 mA	200 mA	200 mA	200 mA
Modo fuente	150 mA	200 mA	200mA	200mA

TABLA 2.0 CAPACIDAD MÁXIMA DE CORRIENTE TOTAL DE LOS PUERTOS DEL MICROCONTROLADOR

El consumo de corriente del microcontrolador para su funcionamiento depende del voltaje de operación, la frecuencia y de las cargas que tengan sus pines. Para un oscilador de 4 MHz el consumo es de aproximadamente 2 mA; aunque este se puede reducir a 40 microamperios cuando se está en el modo sleep (en este modo el micro se detiene y disminuye el consumo de potencia).

2.1.7.4 DESCRIPCIÓN DE LOS PINES DEL MICROCONTROLADOR

NOMBRE PIN	PIN	DESCRIPCIÓN
RA0/AN0	2	E/S Digital o Entrada análoga 0.
RA1/AN1	3	E/S Digital o Entrada análoga 1.
RA2/AN2 Vref -	4	E/S Digital o Entrada análoga 2.
RA3/AN3/Vref +	5	E/S Digital o Entrada análoga 3.
RA4/T0CKI	6	Bit 4 del puerto A (E/S bidireccional). También se usa como entrada de reloj al temporizador/contador TMR0.
RA5/SS/AN4	7	E/S Digital o Entrada análoga 4. También lo usa el puerto serial síncrono.
RB0/INT	33	Bit 0 del puerto B (E/S bidireccional). Buffer E/S: TTL/ST. También se usa como entrada de interrupción externa (INT).
RB1	34	Bit 1 del puerto B (E/S bidireccional). Buffer E/S: TTL
RB2	35	Bit 2 del puerto B (E/S bidireccional). Buffer E/S: TTL
RB3/PGM	36	Bit 3 del puerto B (E/S bidireccional). Buffer E/S: TTL (Programación en bajo voltaje)
RB4	37	Bit 4 del puerto B (E/S bidireccional). Buffer E/S: TTL. Interrupción por cambio del pin.
RB5	38	Bit 5 del puerto B (E/S bidireccional). Buffer E/S: TTL. Interrupción por cambio del pin.
RB6/PGC	39	Bit 6 del puerto B (E/S bidireccional). Buffer E/S: TTL/ST. Interrupción por cambio del pin. Entrada de reloj para programación serial
RB7/PGD	40	Bit 7 del puerto B (E/S bidireccional). Buffer E/S: TTL/ST. Entrada de datos para programación serial.
RC0/T1OSO/T1CKI	15	E/S Digital. Salida del oscilador Timer1 o entrada de reloj Timer1.
RC1/T1OSI/CCP2	16	E/S Digital. Entrada del oscilador Timer 1. Entrada Captura 2; Salida Compara 2; Salida PWM 2
RC2/CCP1	17	E/S Digital. Entrada Captura 1; Salida Compara 1; Salida PWM 1
RC3/SCK/SCL	18	E/S Digital. Línea de reloj serial asíncrono en el modo SPI y el modo I^2C
RC4/SDI/SDA	23	E/S Digital. Línea de datos en el modo SPI o en el modo I^2C
RC5/SDO	24	E/S Digital.
RC6/TX/CK	25	E/S Digital. Transmisión asíncrona (USART) o reloj síncrono (SSP).

TABLA 2.1 DESCRIPCIÓN DE PÍNES DEL MICROCONTROLADOR 16F877A

RC7/RX/DT	26	E/S Digital. Recepción asíncrona (USART) o línea de datos (SSP).
VDD	11,32	Voltaje de alimentación DC (+)
VSS	12,31	Referencia de voltaje (GND).
MCLR	1	Entrada de RESET al microcontrolador. Voltaje de entrada durante la programación. En nivel bajo resetea el microcontrolador.
OSC1/CLKIN	13	Entrada oscilador cristal oscilador / Entrada fuente de reloj externa.
OSC2/CLKOUT	14	Salida oscilador cristal. Oscilador RC: Salida con un ¼ frecuencia OSC1
RD0/PSP0	19	E/S Digital. Puede ser puerto paralelo en bus de 8 bits.
RD1/PSP1	20	E/S Digital. Puede ser puerto paralelo en bus de 8 bits.
RD2/PSP2	21	E/S Digital. Puede ser puerto paralelo en bus de 8 bits.
RD3/PSP3	22	E/S Digital. Puede ser puerto paralelo en bus de 8 bits.
RD4/PSP4	27	E/S Digital. Puede ser puerto paralelo en bus de 8 bits.
RD5/PSP5	28	E/S Digital. Puede ser puerto paralelo en bus de 8 bits.
RD6/PSP6	29	E/S Digital. Puede ser puerto paralelo en bus de 8 bits.
RD7/PSP7	30	E/S Digital. Puede ser puerto paralelo en bus de 8 bits.
RE0/RD/AN5	8	E/S Digital. Puede ser pin de lectura (read) en modo microprocesador.
RE1/WR/AN6	9	E/S Digital. Puede ser pin de escritura (write) en modo microprocesador.
RE2/CS/AN7	10	E/S Digital. Puede ser pin de selección de chip (chip select) en modo microprocesador.

TABLA 2.2 CONTINUACIÓN DE LA DESCRIPCIÓN DE PINES DEL MICROCON 16F877A

2.1.7.5 ARQUITECTURA INTERNA DEL MICROCONTROLADOR

Este término se refiere a los bloques funcionales internos que conforman el microcontrolador y la forma en que están conectados, por ejemplo la memoria FLASH (de programa), la memoria RAM (de datos), los puertos, la lógica de control que permite que todo el conjunto funcione, etc.

La figura 2.4 muestra la arquitectura general del PIC16F877, en ella se pueden apreciar los diferentes bloques que lo componen y la forma en que se conectan. Se muestra la conexión de los puertos, las memorias de datos y de programa, los bloques especiales como el Watchdog, los temporizadores de arranque, el oscilador, etc.

Todos los elementos se conectan entre sí por medio de buses. Un bus es un conjunto de líneas que transportan información entre dos o más módulos. Vale la pena destacar que el PIC16F877 tiene un bloque especial de memoria de datos de 256 bytes del tipo EEPROM, además de los dos bloques de memoria principales que son el de programa y el de datos o registros.

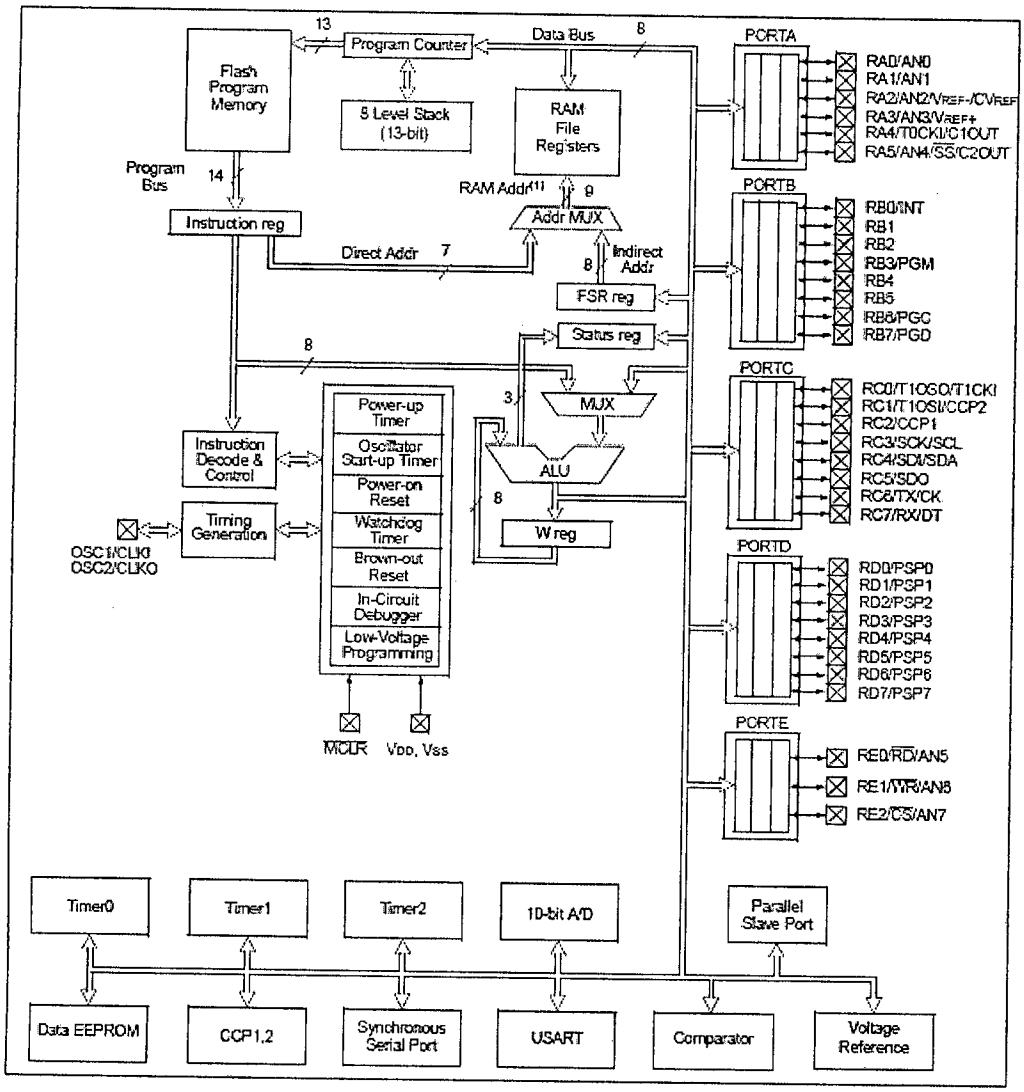


FIG. 2.4 DIAGRAMA DE BLOQUES DEL MICROCONTROLADOR 16F877A

El PIC16F877 se basa en la arquitectura Harvard, en la cual el programa y los datos se pueden trabajar con buses y memorias separadas, lo que posibilita que las instrucciones y los datos posean longitudes diferentes. Esta misma estructura es la que permite la superposición de los ciclos de búsqueda y ejecución de las instrucciones, lo cual se ve reflejado en una mayor velocidad del microcontrolador.

2.1.8 MEMORIA DE DATOS (RAM)

El PIC16F877 posee cuatro bancos de memoria RAM, cada banco posee 128 bytes. De estos 128 los primeros 32 (hasta el 1Fh) son registros que cumplen un propósito especial en el control del microcontrolador y en su configuración. Los 96 siguientes son registros de uso general que se pueden usar para guardar los datos temporales de la tarea que se está ejecutando, figura 2.5 Todas las posiciones o registros de memoria se pueden acceder directa o indirectamente (esta última forma a través del registro selector FSR). Para seleccionar que página o banco de memoria se trabaja en un momento determinado se utilizan los bits RP0 y RP1 del registro STATUS.

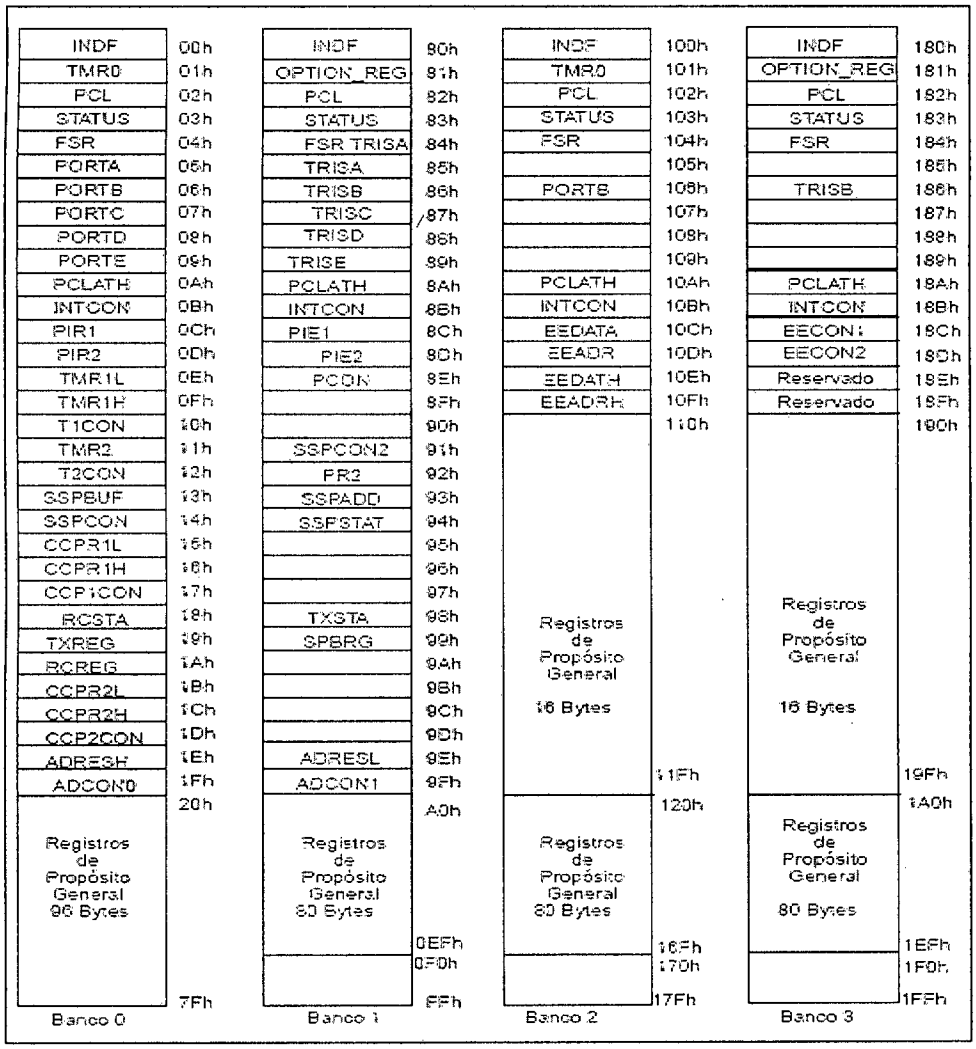


FIG. 2.5 ORGANIZACIÓN DE LA MEMORIA RAM DEL PIC16F877

2.1.8.1 RESUMEN DE ALGUNOS DE LOS REGISTROS DE CONFIGURACIÓN

BANCO 0:

TMR0: Registro del temporizador/contador de 8 bits.

PCL: Byte menos significativo del contador de programa (PC).

STATUS: Contiene banderas (bits) que indican el estado del procesador después de una operación aritmética/lógica.

FSR: Registro de direccionamiento indirecto.

PORTA, PORTB, PORTC, PORTD, PORTE: Registro de puertos de E/S de datos. Conectan con los pines físicos del micro.

PCLATH: Byte alto (más significativo) del contador de programa (PC).

INTCON: Registro de control de las interrupciones.

ADRESH: Parte alta del resultado de la conversión A/D.

ADCON0: Controla la operación del módulo de conversión A/D

BANCO 1:

OPTION: Registro de control de frecuencia del TMR0.

TRISA, TRISB, TRISC, TRISD. TRISE: Registros de configuración de la operación de los pines de los puertos.

ADRESL: Parte baja del resultado de la conversión A/D.

ADCON1: Controla la configuración de los pines de entrada análoga.

BANCO 2:

TMR0: Registro del temporizador/contador de 8 bits.

PCL: Byte menos significativo del contador de programa (PC).

FSR: Registro de direccionamiento indirecto.

EEDATA: Registro de datos de la memoria EEPROM.

EEADR: Registro de dirección de la memoria EEPROM.

PCLATH: Byte alto (más significativo) del contador de programa (PC).

INTCON: Registro de control de las interrupciones.

BANCO 3:

OPTION: Registro de control de frecuencia del TMR0.

EECON1: Control de lectura/escritura de la memoria EEPROM de datos.

EECON2: No es un registro

2.2 SEGURIDAD CIUDADANA

La seguridad ciudadana es la acción integrada que desarrolla el Estado, con la colaboración de la ciudadanía y de otras organizaciones de bien público, destinada a asegurar su convivencia pacífica, la erradicación de la violencia, la utilización pacífica y ordenada de vías y de espacios públicos y, en general, evitar la comisión de delitos y faltas contra las personas y sus bienes.

En los países hispano hablantes hay ocasiones en las que se prefiere usar términos como 'orden público' o 'seguridad de los habitantes' en vez de 'seguridad ciudadana o seguridad nacional', por motivos históricos que dependen de cada país.

En líneas generales, por 'seguridad ciudadana' deben entenderse el conjunto de acciones democráticas en pro de la seguridad de los habitantes y de sus bienes, y ajustadas al derecho de cada país. De hecho, el reto actual es armonizar el ejercicio de los derechos humanos de cada uno con las distintas políticas en materia de seguridad ciudadana de los estados. Por ejemplo, la Organización de los Estados Americanos plantea que en ocasiones se aplican políticas que se han demostrado ineficaces, como por ejemplo el aumento de las penas, la reducción de garantías procesales, o medidas para aplicar el derecho penal a menores de edad; que pueden derivar en movimientos paramilitares o parapoliciales milicias de 'autodefensa' cuando el Estado no es capaz de reaccionar de una forma eficaz ante la violencia y el delito, complicando la situación.

2.2.1 SEGURIDAD CIUDADANA EN EL PERU

Situación Actual

Existen múltiples indicadores para medir la situación de inseguridad, la violencia y el delito en un determinado territorio. Los más importantes son los homicidios, la victimización, la percepción de inseguridad, la confianza en las instituciones y la situación del sistema penitenciario.

a) Homicidios

El Observatorio de Criminalidad del Ministerio Público registra que, entre los años 2009 y 2012, fueron asesinadas 512 mujeres en un contexto de feminicidio en el país. El 73.0 % fueron cometidos por la pareja o ex pareja, el 14.5 % por algún familiar, el 6.8 % por un conocido, el 4.1 % por un desconocido que atacó sexualmente a la víctima y el 1.6% por el cliente de una trabajadora sexual. Más de la mitad de las víctimas (52.3 %) tenía entre 18 y 34 años de edad, rango que constituye el grupo etario más vulnerable. El último año, Madre de Dios registró la tasa más alta de feminicidio (9.2 por 100 mil habitantes mujeres), seguido de Tacna (3.8).

b) Victimización

La victimización es un indicador que mide la ocurrencia real de hechos de violencia o de despojo. Es el más útil para conocer la magnitud de los niveles delictivos, especialmente aquellos de naturaleza patrimonial. Se mide a través de las estadísticas oficiales y de las encuestas de opinión pública. La ventaja de estas últimas radica en que no todos los delitos son denunciados ante las autoridades. En todo caso, es siempre aconsejable complementar el análisis con ambas fuentes de información.

Los primeros años de los noventa (1990-1995) tuvieron la tasa de denuncias de delitos presentadas por el público ante la Policía por 100 mil habitantes más alta de los últimos veintitrés años (902.2), con picos que llegaron a los 1170 y

1255 los años 1992 y 1993. Desde entonces, la tasa fue cayendo significativamente hasta llegar a 506 el año 2007, la más baja de todo el período, para volver a subir de manera sostenida y llegar a los 846 el 2012 , la cuarta más alta.

c) Percepción de inseguridad

Una de las formas de medir la percepción de inseguridad es consultando a las personas sobre el temor que sienten frente a la posibilidad de ser víctimas de un delito en el futuro. Al respecto, el Barómetro de las Américas da cuenta que en esta materia el Perú ha expresado una mejoría en los últimos años, pasando de un 60.0 % el 2006 al 48.6 % el 2012.

A continuación se muestran algunos cuadros:

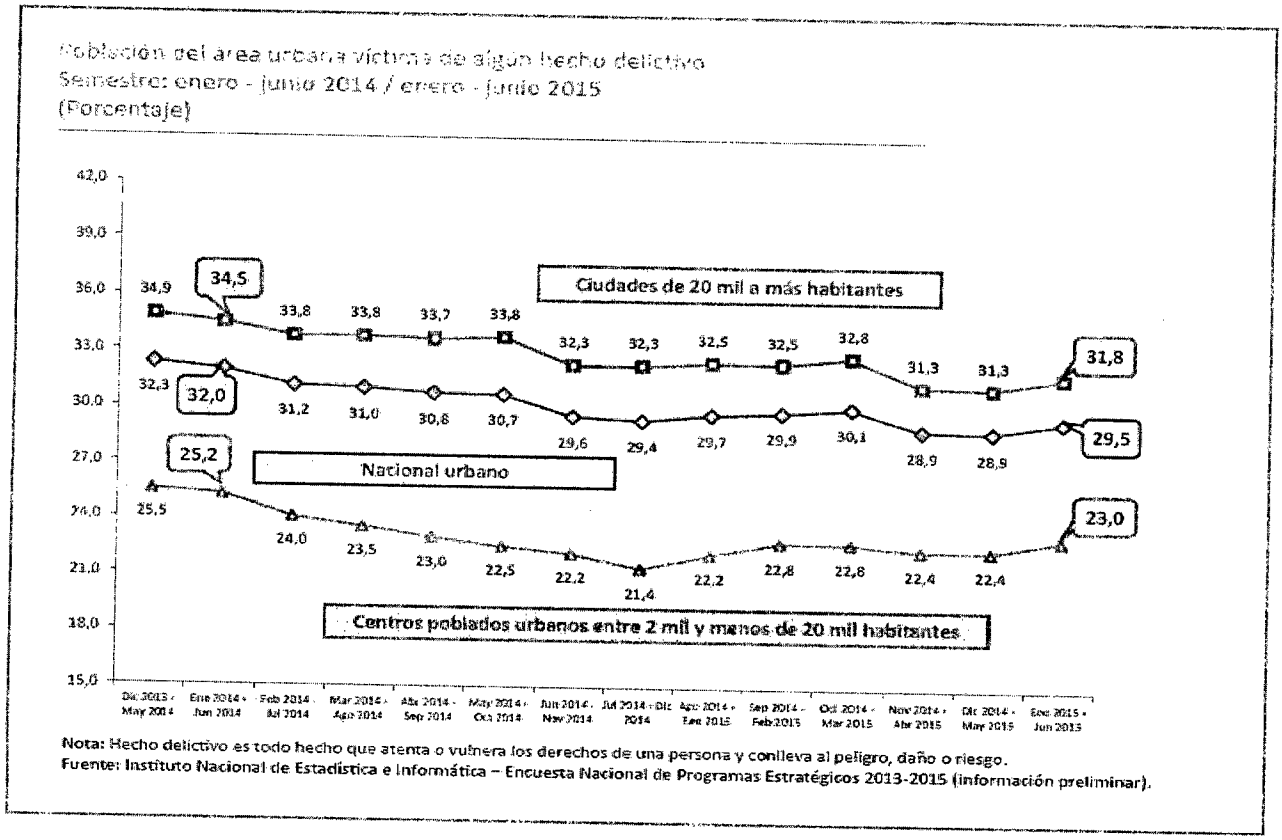


Figura 2.6 Cuadro Población del área urbana víctima de algún hecho delictivo

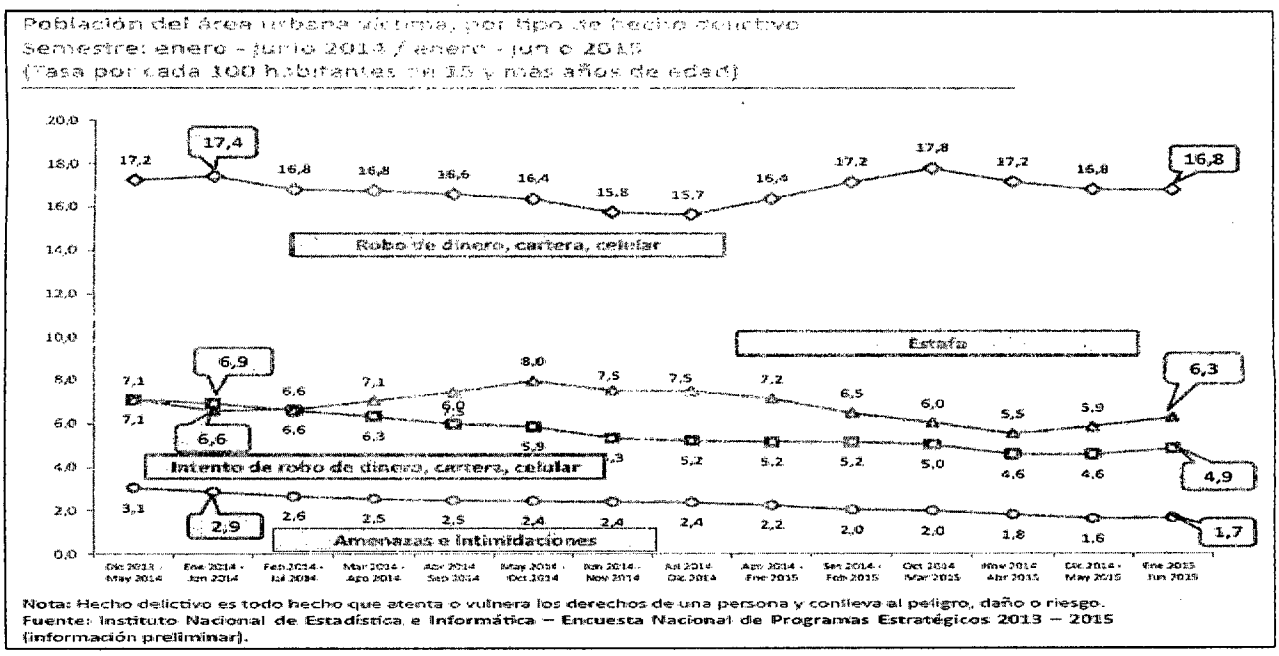


Figura 2.7 Cuadro Población del área urbana víctima por tipo de hecho delictivo

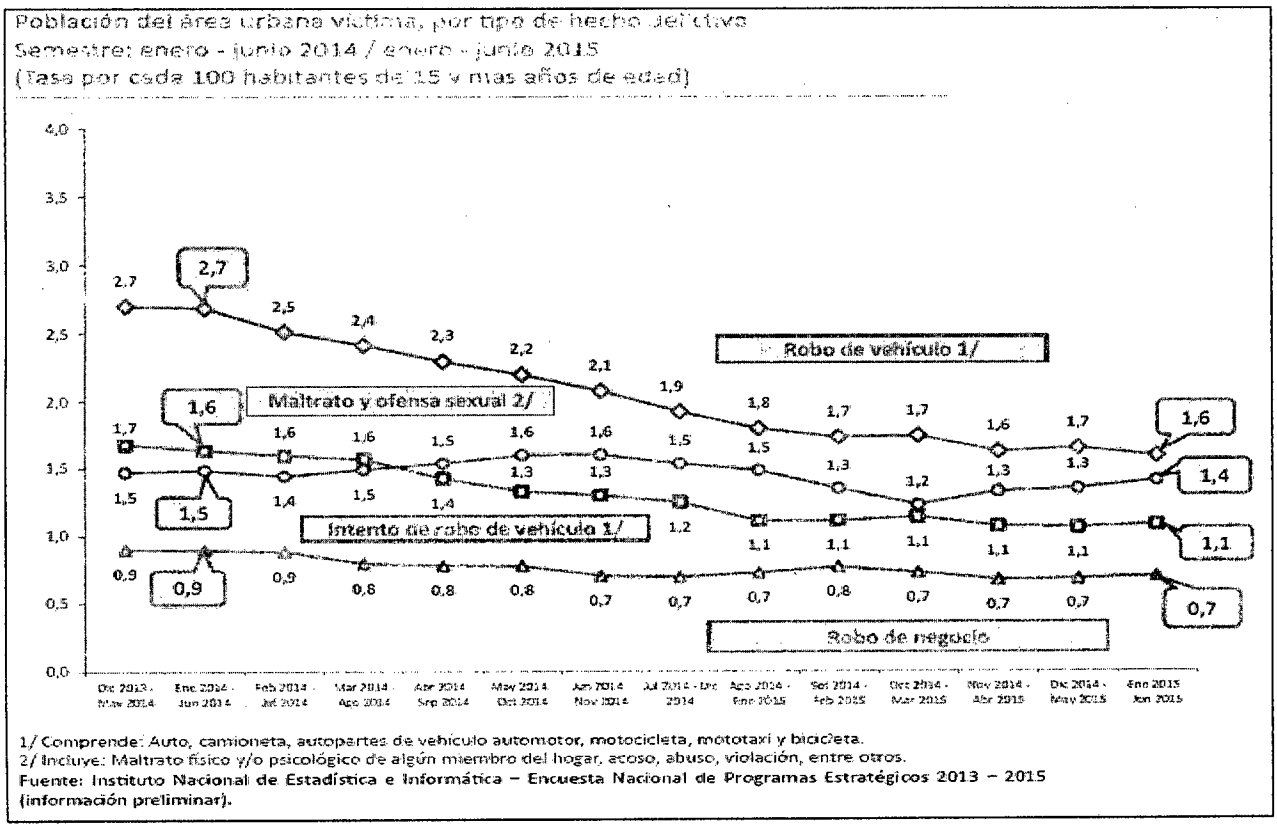


Figura 2.8 Cuadro Población del área urbana víctima por tipo de hecho delictivo

2.3 COMISARIA

Dependencia policial encargada de mantener el orden público, con funciones preventivas y de investigación en una determinada jurisdicción a nivel nacional.

Tipos de Comisarias

Comisaría básica

Es aquella que se encuentra tipificada en A, B, C, D y E de acuerdo al número de efectivos policiales, densidad poblacional, servicios requeridos y área mínima requerida de construcción. Dicha tipificación está debidamente reglamentada por la Policía Nacional del Perú.

Comisaría especializada

Son aquellas que desarrollan un servicio específico, comprende comisarías de mujeres (CAVIFAN), turismo, aeropuertos, terminales terrestres y protección de carreteras.

Comisaría PNP tipo A

Con capacidad de 121 a 240 efectivos policiales. Hasta 1,020 m2 de área construida. Cobertura de 80,001 a 160,000 habitantes.

Comisaría PNP tipo B

Con capacidad de 61 a 120 efectivos policiales. Hasta 680 m2 de área construida. Cobertura de 40,001 a 80,000 habitantes.

Comisaría PNP tipo C

Con capacidad de 31 a 60 efectivos policiales. Hasta 415 m2 de área construida. Cobertura de 20,001 a 40,000 habitantes.

Comisaría PNP tipo D

Con capacidad de 16 a 30 efectivos policiales. Hasta 285 m2 de área construida. Cobertura de 10,001 a 20,000 habitantes.

Comisaría PNP tipo E

Con capacidad de 8 a 15 efectivos policiales. Hasta 245 m2 de área construida. Cobertura de 5,000 a 10,000 habitantes.

2.4 POLICÍA

Se denomina policía a la persona encargada de mantener el orden público y cuidar de la seguridad ciudadana. El policía es uno de los usuarios finales del sistema.

2.5 DISPOSITIVO MÓVIL

Es un dispositivo de cómputo diminuto que también se conoce como dispositivo de mano, portátil o computadora de mano. Los dispositivos móviles suelen venir con una pantalla táctil o no táctil y a veces, incluso un mini teclado.

2.5.1 TELEFONIA MOVIL EN AMERICA LATINA Y EL PERU

La evolución del ecosistema móvil en los últimos 10 años ha generado un proceso acelerado de adaptación de dispositivos móviles en el mundo.

En la actualidad se estima que existen igual cantidad de dispositivos móviles que habitantes en el planeta. Según la ITU (International Telecommunication Union) de las Naciones Unidas, a principios del 2013 existían 6.8 billones de equipos celulares. Del mismo modo, en los 2 últimos años se ha experimentado un explosivo avance en la penetración de dispositivos móviles inteligentes (smartphones). Se calcula que en la actualidad existen 1.2 billones de smartphones en el mundo Igualmente desde el cuarto trimestre del 2010, las ventas de smartphones y tablets superaron a las ventas de PCs. La aparición

del iPhone, el sistema operativo Android de Google y la ampliación de las redes de banda ancha móvil han generado una "tormenta perfecta" para que el mercado de smartphones alcance niveles de crecimiento muy altos a nivel mundial. Otro punto importante que nos da luces sobre esta revolución es que a julio del 2013, el 17% del tráfico mundial de Internet venía desde dispositivos móviles (sin incluir tablets), un crecimiento de 56% vs julio del año 2012, . Igualmente se estima que para el año 2015, el tráfico de Internet proveniente de dispositivos móviles alcanzará al del resto de dispositivos existentesiii. Será el mundo del Internet móvil como ya lo describen algunos analistas. Conectados en todo momento y en todo lugar. Esta es una realidad, ya no del futuro sino del presente. Latinoamérica y el Perú no se escapan de este fenómeno, y como veremos más adelante, sino estamos preparados como empresas, marcas, proveedores de Internet, generadores de contenidos, agencias, medios y demás actores del ecosistema digital, estaremos un paso atrás y perdiendo una gran oportunidad para mantenernos a la vanguardia con el avance de la tecnología.

El mercado móvil en América Latina ha experimentado un fuerte crecimiento en estos últimos años, en el número de suscriptores únicos y conexiones (tarjetas SIM). Actualmente es el cuarto mercado más grande del mundo, con casi 326 millones de suscriptores únicos y 718 millones de conexiones2 en septiembre de 2014. La base de suscriptores creció a un ritmo medio de 5.5% durante los cinco años hasta 2013, y se prevé que crezca a un ritmo ligeramente inferior al 3% anual en el periodo entre 2013 y 2020. El crecimiento de los conexiones en el periodo hasta 2013 fue de un 8.8% anual, y las previsiones indican que disminuirá a apenas encima del 3.7% de media en el periodo de siete años hasta 2020.

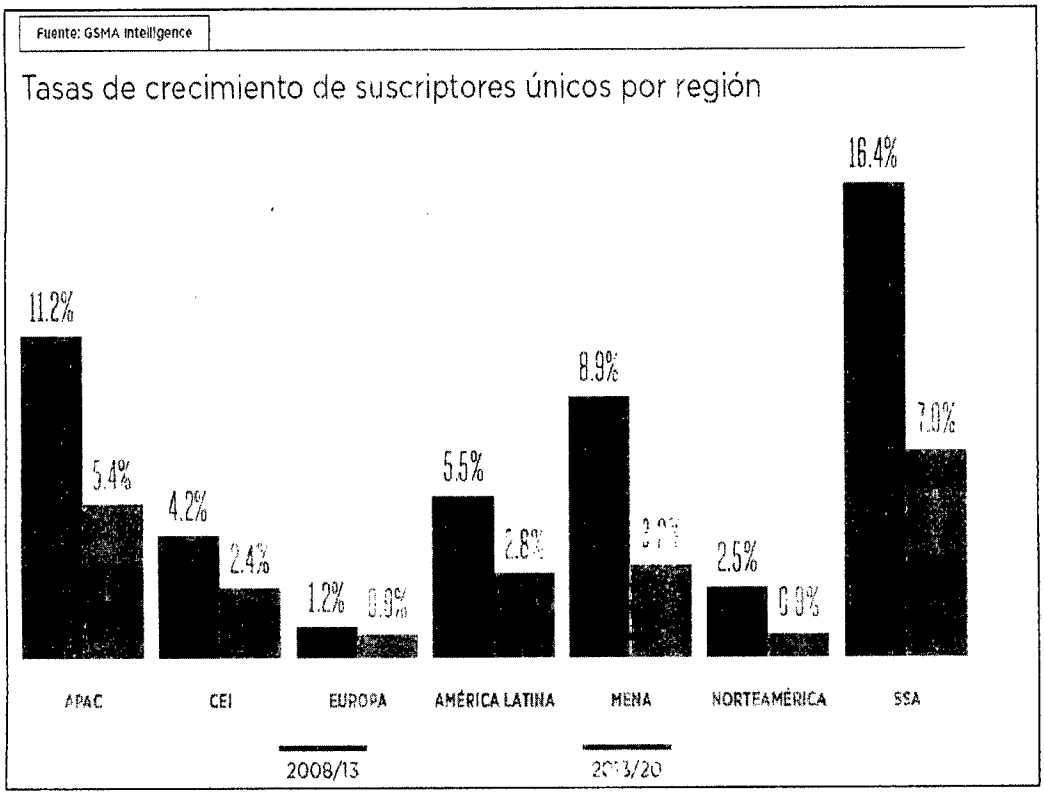


Figura 2.9 Tasa de crecimiento de suscriptores únicos por región

América Latina es una región muy diversa en cuanto al desarrollo social y económico. También lo son los niveles de penetración móvil (tanto de suscriptores únicos como de conexiones). Las tasas de penetración de conexiones van desde el nivel más bajo del 73% en Haití hasta el más alto del 157% en Costa Rica. La tasa general de penetración en toda la región era del 112% en septiembre de 2014, muy por encima de la media mundial del 96%.

Las tasas de penetración de suscriptores en los mercados más grandes van desde el nivel más bajo del 37%, en México, hasta el más alto: un 77% en Costa Rica. No existe un factor único que motive la variación en las tasas de penetración; por ejemplo, las diferencias en el PIB per cápita solo tienen una influencia limitada.

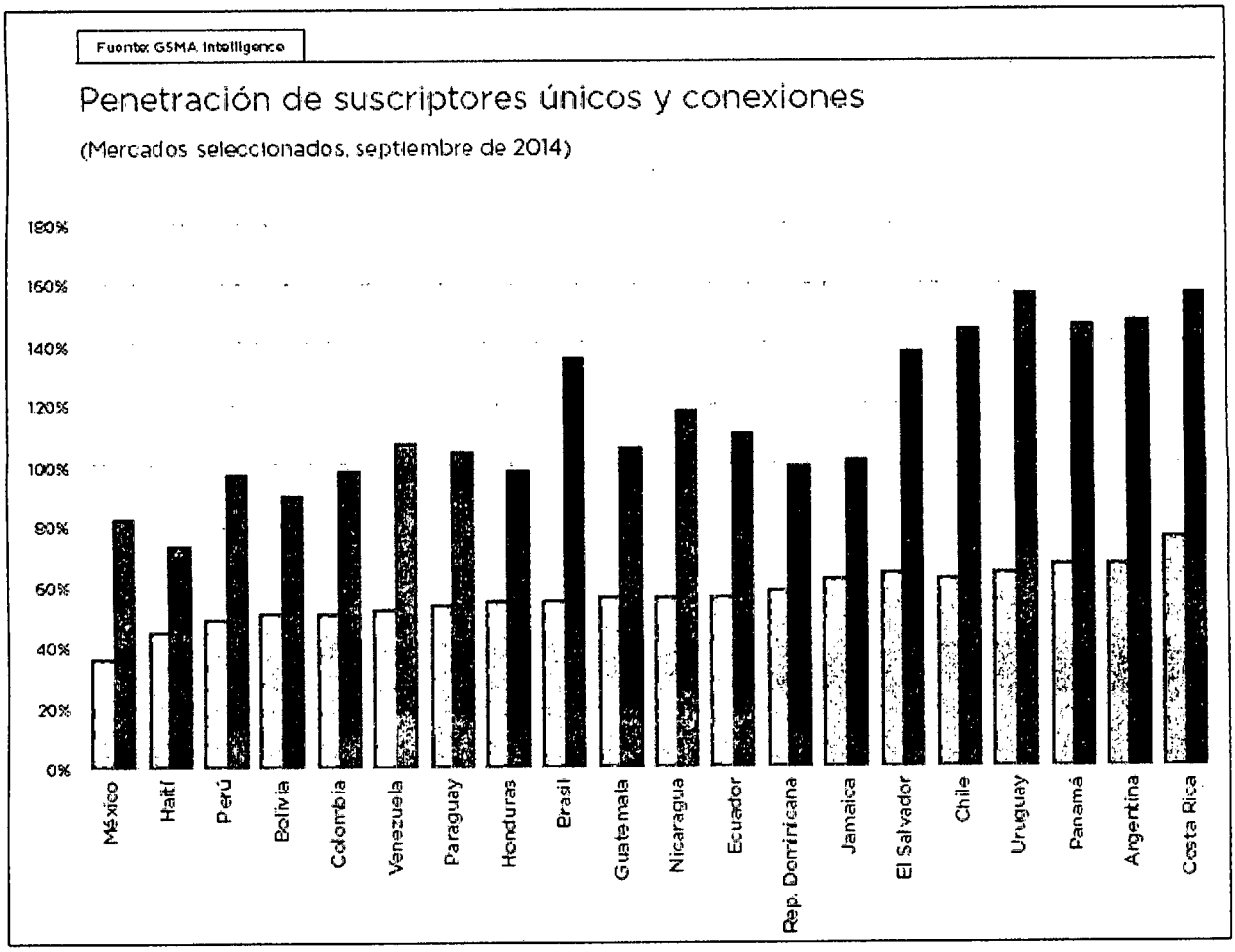


Figura 2.10 Penetración de suscriptores únicos y conexiones

2.5.2 APLICACIONES MOVILES Y SEGURIDAD CIUDADANA

Una nueva aplicación para teléfonos inteligentes de Brasil mapea los lugares en donde ocurrieron varios tipos de crímenes y ofrece a los usuarios la ruta más segura para llegar a casa. Este es sólo un ejemplo de cómo se está utilizando la tecnología proporcionada por los dispositivos móviles para mejorar la seguridad ciudadana en Latinoamérica.

La aplicación, llamada Smaps, permite a los usuarios reportar el lugar en donde los crímenes -que van desde el asesinato y el secuestro al vandalismo- están ocurriendo en tiempo real, y notifica la actualización a otros usuarios registrados. Al usar esta base de datos de información compartida, la aplicación puede trazar la ruta más segura hacia el destino final de un usuario determinado, y también puede identificar la estación de policía más cercana.

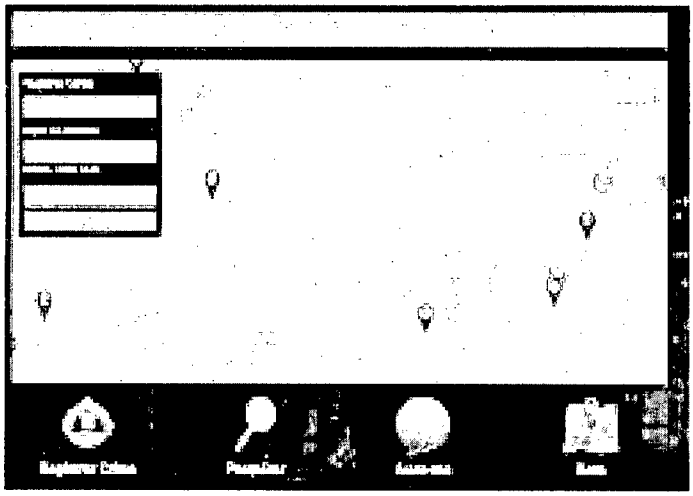


Figura 2.11 Pantalla de SMAPS

Aunque Smaps fue desarrollada en Brasil, esta aplicación puede ser utilizada a nivel internacional. La aplicación actualmente tiene la mayor cantidad de datos en grandes ciudades como Río de Janeiro y São Paulo, según O Globo. La persona que desarrolló esta aplicación, Douglas Roque, al parecer tiene previsto ampliar la base de datos del sitio mediante la inclusión de información sobre el crimen, disponible públicamente, de fuentes gubernamentales locales, además de delitos reportados por los ciudadanos.

Smaps es sólo una aplicación de teléfono inteligente destinada a mejorar la seguridad ciudadana en Brasil. La aplicación "CopCast", que forma parte de la iniciativa Smart Policing desarrollada por el centro de estudios brasileño Instituto Igarapé, permite a los funcionarios experimentados de la policía monitorear a los oficiales de patrulla mediante el uso de un mapa interactivo y en tiempo real. CopCast también puede almacenar videos de las cámaras del cuerpo policial durante un máximo de 90 días.

Dos metas de Smart Policing son aumentar la rendición de cuentas por parte de los oficiales policiales que usan la fuerza excesiva y la protección de la policía a las acusaciones infundadas de abusos. La Policía Militar de Río de Janeiro es un socio en el proyecto, el cual actualmente está siendo probado con las Unidades de Policía Pacificadora (UPP) en Río.

El Instituto Igarapé también está desarrollando el Índice de Seguridad Infantil (CSI, por sus siglas en inglés), el cual, a través de una aplicación para teléfonos inteligentes, "mapea espacialmente y visualiza gráficamente cómo los jóvenes sufren la violencia en los barrios pobres" de Brasil. El CSI logra esto al usar la aplicación para encuestar a los niños sobre sus percepciones de peligro y, con base en sus respuestas, crear gráficos y mapas interactivos los cuales identifican los lugares en donde los niños se sienten en riesgo (vea la imagen abajo).

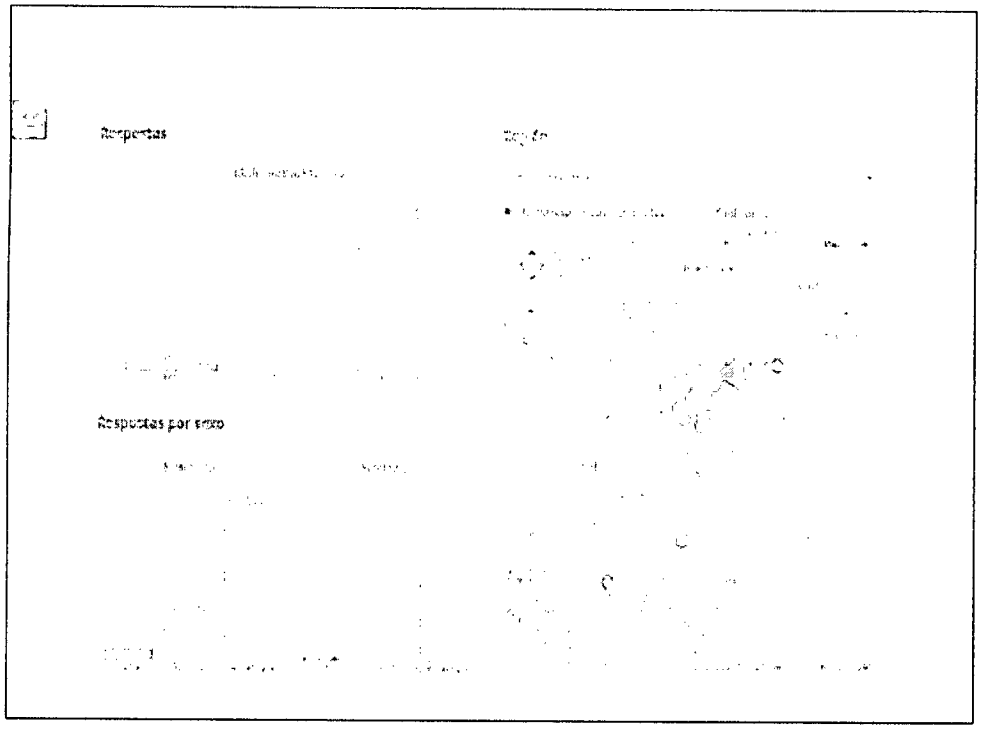


Figura 2.12. Pantalla del CSI

Según Robert Muggah, el Director de Investigación y Coordinador del Programa de Seguridad Ciudadana en Igarapé, el CSI busca llenar un "vacío de conocimiento" sobre cómo los niños menores de 10 años sufren la violencia. También se espera que el CSI llegue a medir la efectividad de los programas de intervención en jóvenes que promuevan la seguridad de los niños -como las Escuelas de Paz en Río- midiendo periódicamente la manera como las

respuestas a la encuesta por parte de los niños en estos programas cambian con el tiempo.

Las aplicaciones de seguridad para los teléfonos inteligentes están ganando terreno en muchos otros lugares de Latinoamérica, además de Brasil. Aplicaciones que actualmente están disponibles en países como México, Venezuela, Perú y Colombia pueden conectar a los usuarios de forma instantánea con estaciones de policía en situaciones de emergencia. Estas aplicaciones ya están mostrando resultados en Perú: el gerente de Seguridad Ciudadana del distrito de Surco, en Lima, informó que la delincuencia bajó 40 por ciento entre enero -cuando la aplicación comenzó a ser utilizada- y marzo de este año.

2.5.3 TENDENCIAS DE LAS APLICACIONES MOVILES

Perfil del Mercado Peruano

Uno de los principales retos que hemos encontrado a la hora de preparar este documento es la escasa información que tenemos sobre la situación actual del mercado móvil peruano. Ante la diversidad de información y fuentes, he decidido concentrarme en las que tienen un mayor nivel de confianza:

1) Penetración de teléfonos móviles (celulares)

De acuerdo al estudio de Ipsos Apoyo del año pasado, "Usos y actitudes hacia la telefonía móvil 2012", en la actualidad existe una penetración de 75% de celulares sobre la población nacional urbana entre 12 y 70 años de edad (18 millones de personas). Esto nos da como resultado un total de 12 millones de equipos celulares en el mercado peruanoiv. Hay que tomar en cuenta que este número difiere de la cantidad de líneas móviles que existen en el mercado que están alrededor de 30 millones, muchas de las cuales están no activas y que tienden a generar un error en el número real de personas que cuentan con teléfono celular.

2) Penetración de dispositivos móviles inteligentes (smartphones)

De acuerdo al mismo estudio de Ipsos Apoyo, la penetración de dispositivos móviles inteligentes es del 16% sobre la base de la población encuestada, es decir 12 millones de personas. Esto nos da como resultado un total de 1.9 millones de equipos smartphones en el mercado peruano. Existe igualmente una encuesta anual de Osiptel sobre demanda del año 2012, en la cual se habla de una penetración de smartphones de 14% sobre el total de usuarios residencialesv. Cifra similar a la de Ipsos Apoyo. Es claro mencionar que ambas cifras se refieren al año 2012, con tasas de crecimiento superiores al 50% por lo que podemos inferir que para fines de este año (2013) no debería haber menos de 3 millones de smartphones en el mercado peruano, representando casi un 25% de penetración.

3) Usos de dispositivos móviles

De acuerdo al estudio de Ipsos Apoyo, dentro de los principales usos que se le da al dispositivo móvil en el Perú (aparte de hacer llamadas), está el mandar mensajes de texto (SMS), chatear a través de aplicativos de mensajería instantánea tipo BBM, WhatsApp, Facebook Chat, etc., y conectarse a una red social tipo Facebook o Twitter. Otros usos adicionales son el de la cámara de fotos y conectarse a Internet para buscar información.

4) Sistemas operativos móviles

Si bien no existen estadísticas oficiales sobre la participación de los sistemas operativos móviles en nuestro país, hay algunos indicadores que nos pueden dar una idea de los porcentajes. Así tenemos los siguientes números referenciales a julio del presente añovi:

- Android: 60%
- iOS: 15%
- BlackBerry: 8%

- Windows Phone: 7%
- Otros: 10%

Es clara la tendencia creciente en participación de Android en el mercado peruano, el estable porcentaje de participación de iOS y la caída fuerte en los 2 últimos años de BlackBerry, siguiendo la tendencia mundial. En realidad todo lo que ha crecido Android es lo que ha perdido BlackBerry. Un aspecto interesante es la aparición de Windows Phone como un nuevo jugador en el mercado. Otro punto importante en la evaluación del mercado actual es la conexión a Internet a través de dispositivos móviles. Según el estudio de Ipsos Apoyo, el 44% del total de poseedores de un equipo celular tiene acceso a Internet (5 millones) y de ellos sólo el 20% la utiliza (2.6 millones). De este último número el 73% tiene un plan de datos o contrata MB de navegación independientemente. Como nos indican estos números, en el Perú todavía estamos atrasados tanto en penetración de celulares en general como de smartphones respecto a países desarrollados. Una de las principales razones para no contar con un equipo móvil es el factor económico, habiendo un 45% de personas que no lo tienen porque le resulta muy caro, según la encuesta anual de Osiptel. Lo mismo sucede con los planes de datos, se necesita una mayor y mejor oferta de los operadores móviles para masificar el acceso al Internet móvil. Debemos tomar en cuenta que en los países menos desarrollados, el Internet móvil se ha vuelto el único canal de acceso a Internet que tienen las personas menos favorecidasvii. El desarrollo de infraestructura es fundamental y en ello deben concentrase los esfuerzos públicos y privados.

5) Uso de los dispositivos móviles como canal de interacción

Conforme hemos ido presentando los datos anteriores, vamos visualizando el gran impacto que los dispositivos móviles están teniendo en la población mundial y en el país. Esta revolución, como no podía ser de otra manera, está impactando el marketing y la publicidad. Conforme la gente pasa más tiempo

con su dispositivo móvil, interactúa más con él y se conecta más tiempo a Internet, más se puede aprovechar para poder impactar en clientes y potenciales clientes.

Sin embargo, todavía hay una gran brecha a nivel mundial entre lo que se invierte en marketing móvil versus el tiempo que cada vez más le dedican las personas a esta herramienta. Según un estudio de Kleiner, Perkins, Caufield, Byers, existe una brecha de $20 billones entre el valor de lo que se invierte en publicidad móvil versus el valor del usuario móvil (determinado por el tiempo que le dedica al medio)

Si bien la primera revolución ha estado dada por los dispositivos no inteligentes "feature phones", la verdadera revolución viene del lado de los smartphones. Ahora, qué herramientas tenemos a la mano dentro del marketing móvil. Aquí un listado de las principales:

- SMS: mensajes de texto, usados tradicionalmente en los dispositivos móviles no inteligentes.
- MMS: permite enviar mensajes con contenido multimedia: fotos, videos, etc.
- Web Móvil: contenido web optimizado para dispositivos móviles.
- Aplicaciones (Apps): programas que permiten ser instaladas en el dispositivo móvil.
- Display Ads Móviles: avisos y banners optimizados para dispositivos móviles.
- Realidad Aumentada: aplicaciones que permiten combinar la realidad física con la realidad virtual.
- Códigos QR: códigos que permiten enlazar el mundo off-line con el mundo on-line.
- Redes Sociales: cada vez más gente se conecta a redes sociales a través de su dispositivo móvil. Todas estas herramientas nos permiten interactuar como usuarios. Poder buscar información a través de una web móvil,

escanear un código QR en un medio impreso para ver un video o instalarnos una aplicación para poder ubicar los locales cercanos de un restaurante. Todas estas interacciones están cambiando la forma en que las personas tienen contacto con una marca, un producto o un servicio.

Es por ello que las empresas y las marcas deben comenzar a considerar la inversión en herramientas digitales móviles, el público está migrando a este medio, dejando tiempo que le dedican a otros medios. En entrevistas recientes que hicimos en mi empresa a usuarios de smartphones, el 85% de los entrevistados nos dijo que lo primero que hacía al levantarse era revisar su smartphone. Es una realidad que los ejecutivos de marketing no pueden dejar de lado, si no es hoy, en 2 años esto será el común denominador. Es por ello que se tiene que comenzar a actuar hoy día.

Es fundamental que las empresas comiencen a desarrollar estrategias de marketing móvil y comenzar a integrarlo dentro del "mix" de herramientas que están utilizando para alcanzar a su público objetivo.

2.6 SISTEMA GPRS

2.6.1 General Packet Radio Service (GPRS)

El Servicio General de Paquetes por Radio o GPRS por su acrónimo en inglés (General Packet Radio Service) es una nueva tecnología que comparte el rango de frecuencias de la red GSM utilizando una transmisión de datos por medio de 'paquetes'. La conmutación de paquetes es un procedimiento más adecuado para transmitir datos, hasta ahora los datos se habían transmitido mediante conmutación de circuitos, procedimiento más adecuado para la transmisión de voz [3].

GSM es la tecnología de segunda generación más exitosa celular, pero la necesidad de mayores velocidades de datos dio lugar a nuevos avances para que los datos sean transferidos a un ritmo mucho más alto. El primer sistema para hacer un impacto en el mercado fue GPRS. El GPRS letras corresponden

a General Packet Radio System, la tecnología GPRS a habilitado unas tasas de datos que se transmiten por una red celular en comparación con GSM.

GPRS se convirtió en el primer escalón del camino entre las redes GSM de segunda generación en tecnología celular y el 3G W-CDMA / UMTS. Con la tecnología GPRS los datos que ofrece servicios con tarifas de datos hasta un máximo de 172 kbps, instalaciones como la navegación web y otros servicios que requieren la transferencia de datos se hizo posible. A pesar de que algunos datos pueden ser transferidos a través de GSM, la tasa fue demasiado lento para aplicaciones de datos en tiempo real [4].

2.6.1.1 Arquitectura GPRS

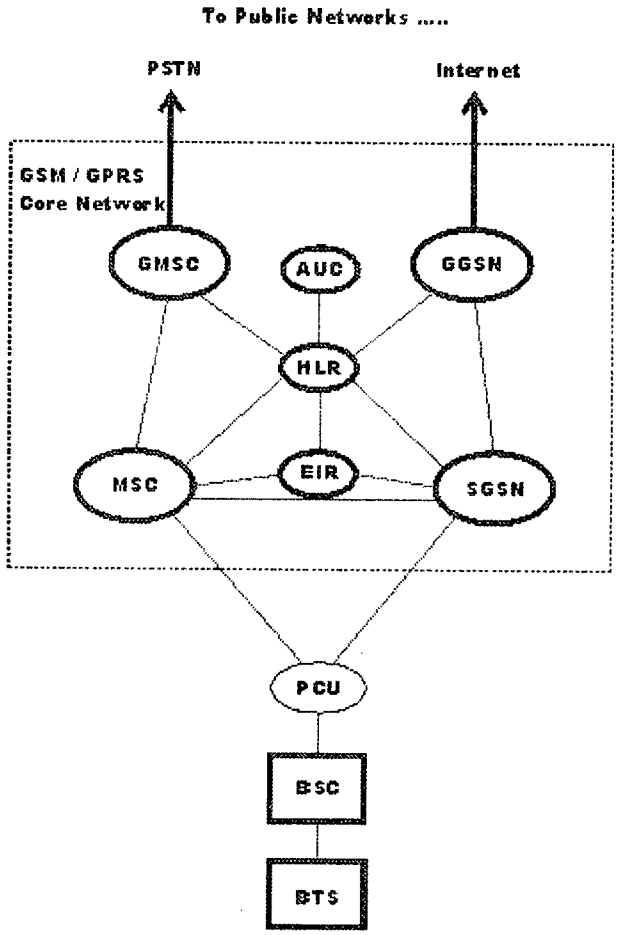
Con GPRS proporciona conectividad adicional en términos de paquetes de datos, hay, naturalmente, una serie de mejoras necesarias para la arquitectura de red utilizada. Una serie de nuevos elementos son necesarios para la red, pero puede funcionar junto con los elementos existentes en el sentido de que la capacidad GPRS es una actualización a la red y no una estructura de red completamente nueva.

Las principales entidades en nueva arquitectura de red que se necesitan son:

- SGSN: GPRS Support Node - esta forma una puerta de acceso a los servicios dentro de la red.
- GGSN: GPRS Gateway Support Node que constituye la puerta de entrada al mundo exterior.
- UCP: Unidad de control de paquetes que diferencia si los datos se enrutan a la conmutación de paquetes o redes de conmutación de circuitos.

Una visión simplificada de la arquitectura de la red GPRS se puede ver en la esquema 1. Se observa que es muy similar a la arquitectura de red GSM más

básico, pero con elementos adicionales. Esquema 1. Arquitectura de la red GPRS



Esquema 1 RED GPRS

2.6.1.1.1 SGSN

El SGSN o porción nodo de soporte de GPRS, es elemento de la red GPRS que ofrece una serie de tomas, se centra en los elementos de propiedad intelectual de todo el sistema. Proporciona una variedad de servicios para los móviles:

- El enrutamiento de paquetes y la transferencia
- Gestión de la movilidad
- Conexión / desconexión
- Lógica de gestión de enlace

- Autenticación
- La carga de datos

Hay un lugar en el registro de esta información SGSN y los puntos de venta (por ejemplo, la celda actual, VLR actual). También almacena los perfiles de usuario (por ejemplo, IMSI, paquete de direcciones utilizadas) para todos los usuarios registrados GPRS con el SGSN en particular [4].

2.6.1.1.2 GGSN

El GGSN, GPRS Gateway Support Node es una de las entidades más importantes dentro de la arquitectura de la red GPRS.

El GGSN organiza la interoperabilidad entre la red GPRS y redes de conmutación de paquetes externos a los que los móviles se pueden conectar. Estos pueden incluir tanto a Internet y las redes X.25.

El GGSN puede ser considerado como una combinación de una puerta de acceso, router y firewall, ya que oculta la red interna hacia el exterior. En la operación, cuando el GGSN recibe datos dirigidos a un usuario específico, comprueba si el usuario está activo, el reenvío de los datos. En la dirección opuesta de paquetes de datos desde el móvil se dirige a la red de destino junto al GGSN.

2.6.1.1.3 UCP

La UCP o del paquete Unidad de Control es un router de hardware que se añade a la BSC. Se diferencia de datos destinado a la red estándar GSM (datos conmutados por circuito) y los datos destinados a la red GPRS (conmutación de paquetes de datos). La propia UCP puede ser una entidad física independiente, o con mayor frecuencia en estos días, se ha incorporado al controlador de estación base BSC, de tal modo de ahorro de los costes adicionales de hardware.

2.6.2 Modulación del GPRS

GPRS se basa en la estructura de base GSM. Utiliza el formato de la señal mismo ancho de banda que tiene 200 kHz. También tiene el mismo esquema de modulación y el uso de la modulación GMSK. Conservando el mismo esquema de modulación, significa que el nivel de actualización necesario para ser capaz de soportar GPRS, además de GSM se reduce al mínimo.

2.6.3 GPRS Categorías de desempeño

No todos los móviles GPRS están diseñados para ofrecer los mismos niveles de servicio. Como resultado, se dividen en tres categorías básicas de acuerdo con sus capacidades en términos de la capacidad de conectarse a los servicios GSM y GPRS:

1. Clase A: - Esta clase describe los teléfonos móviles que pueden conectarse tanto a los servicios GSM y GPRS al mismo tiempo.

2. Clase B: - Estos móviles se puede conectar tanto a los servicios GPRS y GSM, pero que se puede utilizar en un solo servicio a la vez. Clase móvil puede hacer o recibir una llamada de voz o enviar y / o recibir un mensaje de texto durante una conexión GPRS B.Durante las llamadas de voz o mensajes de texto al servicio GPRS está suspendida pero se restablece cuando la llamada de voz o una sesión de SMS se ha completado.

3. Clase C: - Esta clasificación cubre los teléfonos que se pueden conectar a cualquiera de los servicios de GSM o GPRS, pero el usuario tiene que cambiar manualmente entre los dos tipos diferentes.

2.6.4 Protocolos del plano de transmisión

El plano de transmisión es el encargado de proveer la transmisión de los datos del usuario y su señalización para el control de flujo, detección de errores y la corrección de los mismos.

2.6.4.1 GPRS Protocolo Tunneling (GTS)

Es el encargado de transportar los paquetes del usuario y sus señales relacionadas entre los nodos de soporte de GPRS (GSN). Los paquetes GTP contienen los paquetes IP o X.25 del usuario. Por debajo de él, los protocolos estándares TCP o UDP se encargan de transportar los paquetes por la red. Resumiendo, en el Backbone del GPRS tenemos una arquitectura de transporte P/X.25-sobre-GTP-sobre-UDP/TCP-sobre IP.

2.6.4.2 Protocolo Convergencia Dependiente Subred (SNDCP)

Es el encargado de transferir los paquetes de datos entre los SGSN (nodo responsable de la entrega de paquetes al terminal móvil) y la estación móvil. Las funciones que desempeña:

- Multiplexación de diversas conexiones de la capa de red en una conexión lógica virtual de la capa LLC.
- Compresión y descompresión de los datos e información redundante de cabecera.

2.6.4.3 Interferencia de aire

Concierne a las comunicaciones entre la estación móvil y la BSS en los protocolos de las capas física, MAC, y RLC.

Las subcapas RLC/MAC permiten una eficiente multiplexación multiusuario en los canales de paquetes de datos compartidos, y utiliza un protocolo ARQ selectivo para transmisiones seguras a través del interfaz aire. El canal físico dedicado para tráfico en modo paquete se llama PDCH(Packet Data Channel).

En adelante se considerará la capa de enlace de datos (Data Link Layer) y la capa física (Physical Layer) como parte del Interface Aire Um.

2.6.4.4 Capa de enlace de datos

Capa de enlace de datos. Se encuentra entre la estación móvil (el móvil GPRS en sí) y la red.

Dividida en:

- La capa LLC (entre MS-SGSN): Provee un enlace altamente fiable, está basado en el protocolo DIC e incluye control de secuencia, entrega en orden, control de flujo, detección de errores de transmisión y retransmisión. Es básicamente una adaptación del protocolo LAPD usado en GSM.

- La capa RLC/MAC (entre MS-BSS): Incluye dos funciones. El principal propósito de la capa de Control de Radio Enlace (RLC) es la de establecer un enlace fiable. Esto incluye la segmentación y reensamblado de las tramas LLC en bloques de datos RLC y ARQ (peticiones de retransmisión) de códigos incorregibles. La capa MAC controla los intentos de acceder de un MS a un canal de radio compartido por varios MS. Emplea algoritmos de resolución de contenciones, multiplexación de multiusuarios y prioridadesegún la QoS contratada.

2.6.4.5 Capa física

Capa física entre MS y BSS. También se subdivide en dos subcapas.

- La capa del enlace físico (PLL) provee un canal físico. Sus tareas incluyen la codificación del canal (detección de errores de transmisión, corrección adelantada (FEC), indicación de codigos incorregibles), interleaving y la detección de congestión del enlace físico.
- La capa de enlace de radio frecuencia (RFL) trabaja por debajo de la PLL e incluye la modulación y la demodulacion.

2.6.5 Packet Data Protocol (PDP)

Es una estructura de datos presentes tanto en el Nodo de Servicio de Apoyo GPRS (SGSN) y el nodo de pasarela GPRS de apoyo (GGSN) que contiene información del abonado sesión cuando el suscriptor tiene una sesión activa. Cuando un móvil quiere usar GPRS, primero debe conectar y activar un contexto PDP. Esto asigna un contexto PDP estructura de datos en el SGSN que el suscriptor se encuentra de visita y el GGSN sirve punto de acceso del suscriptor. Los datos registrados incluyen

- Suscriptor dirección IP
- Abonado IMSI
- Suscriptor
- Punto final del túnel ID (TEID) en el GGSN
- Punto final del túnel ID (TEID) en el SGSN

El ID de punto final del túnel (TEID) es un número asignado por el GSN, que identifica los datos relacionados con el túnel de un contexto PDP particular.

2.6.6 Acces Point Name (APN)

Es un protocolo informático que permite por lo general el equipo del usuario para acceder a Internet utilizando la red de telefonía móvil. A nivel técnico se trata de un identificador de red configurable utilizado por un dispositivo móvil para conectarse a un GSM operador. El transportista se examinará este identificador para determinar qué tipo de conexión de red debe ser creado, por ejemplo: ¿qué dirección IP debe ser asignado a un dispositivo inalámbrico, lo que los métodos de seguridad se debe utilizar, y cómo / o si, debe estar conectado a alguna red de clientes privados.

Más concretamente, el nombre de punto de acceso (APN) identifica una dirección IP de paquetes de red de datos (PDN), que un usuario móvil de datos quiere comunicar. Además de la identificación de un PDN, una APN también se

puede utilizar para definir el tipo de servicio (por ejemplo, la conexión a Wireless Application Protocol (WAP), servicio de mensajería multimedia (MMS)), que es proporcionada por el PDN. El APN se utiliza en 3GPP redes de acceso de datos, por ejemplo, General Packet Radio Service (GPRS), Evolved Packet Core (EPC).[8].

2.6.7 Point to Point Protocol

En las redes, el protocolo punto a punto, o PPP, es un protocolo de enlace de datos de uso común en el establecimiento de una conexión directa entre dos nodos de red. Puede proporcionar autenticación de la conexión, la transmisión de privacidad cifrada y la compresión.[9].

PPP se utiliza en muchos tipos de redes físicas con cable serie incluido, línea telefónica, la línea troncal, teléfonia celular, enlaces especializados de radio, y enlaces de fibra óptica, tales como SONET. [9]

PPP también se utiliza en conexiones de acceso a Internet (en la actualidad se comercializacomo "banda ancha"). La mayoría de los proveedores de servicios Internet (ISP) se utiliza PPP para el cliente en el acceso telefónico a Internet. Dos formas de encapsulado PPP, Point-to-Point Protocol over Ethernet (PPPoE) y Point-to-Point Protocol sobre ATM (PPPoA), se utilizan más comúnmente por los proveedores de servicios Internet (ISP) para establecer una línea de suscriptor digital (DSL) conexión a Internet de servicios con los clientes.[9]

PPP se utiliza comúnmente como un protocolo de capa de enlace de datos para la conexión a través de circuitos sincrónicos y asincrónicos, donde se ha sustituido en gran medida de la edad del Protocolo Internet de línea serie (SLIP) y la compañía telefónica normas de cumplimiento obligatorio (como el Protocolo de acceso al enlace, equilibrado (LAPB) en la protocolo X.25 suite). PPP fué diseñado para trabajar con múltiples protocolos de capa de

red, incluyendo "Internet Protocol" (IP), "Internetwork Novell Packet Exchange " (IPX), NBF y AppleTalk

2.6.8 Socket

En las redes de computadoras, un Internet socket o network socket es un extremo de una asociación bidireccional entre procesos de flujo de la comunicación a través de un Protocolo de Internet basado en red informática, tales como el Internet.

El término sockets de Internet también se utiliza como un nombre para una interfaz de programación de aplicaciones (API) para el TCP / IP stack de protocolo, por lo general proporcionada por el sistema operativo. Sockets de Internet constituyen un mecanismo para la entrega de paquetes de datos a la aplicación adecuada proceso o hilo, basado en una combinación de locales y remotos direcciones IP y números de puerto. Cada toma se asigna por el sistema operativo a un proceso de solicitud de comunicación o hilo.

Una dirección de socket es la combinación de una dirección IP (la ubicación de la computadora) y un puerto (que se asigna al proceso de programación de aplicaciones) en una sola identidad, al igual que uno de los extremos de una conexión telefónica es la combinación de un número de teléfono y una extensión en particular

2.6.9 Firewall

Es un dispositivo o conjunto de dispositivos diseñados para permitir o denegar las transmisiones de red basado en un conjunto de reglas y se utiliza con frecuencia para proteger las redes del acceso no autorizado al tiempo que permite las comunicaciones legítimas de pasar

Muchos de computadoras personales los sistemas operativos incluyen software basado en servidores de seguridad para proteger contra las amenazas de la Internet pública. Muchos routers que pasar los datos entre las redes contienen

componentes de firewall y, a la inversa, muchos firewalls pueden realizar las funciones básicas de enrutamiento

2.6.10 Protocolo de Transferencia de Archivos (FTP)

Es un protocolo de red estándar que se utiliza para transferir archivos de un host a otro en una red basada en TCP, como el Internet. FTP está construido sobre una arquitectura cliente-servidor y utiliza el control por separado y las conexiones de datos entre el cliente y el servidor. Los usuarios de FTP pueden autenticarse con un inicio de sesión en texto del protocolo, pero puede conectarse de forma anónima si el servidor está configurado para permitirlo

Las primeras aplicaciones de cliente de FTP son interactivos herramientas de línea de comandos, la aplicación de los comandos estándar y la sintaxis. Clientes de interfaz gráficade usuario han sido desarrollados para muchos de los sistemas operativos de escritorio más populares en la actualidad

2.6.11 Protocolo de Control de Transmisión (TCP)

Es uno de los principales protocolos de la suite de protocolo de Internet. TCP es uno de los dos componentes originales de la serie, complementando el Protocolo Internet (IP), y por lo tanto, todo el conjunto se conoce comúnmente como TCP / IP. TCP ofrece una entrega confiable, ordenada de un flujo de bytes de un programa en un ordenador a otro programa en otro ordenador

TCP es el protocolo que las principales aplicaciones de Internet como el World Wide Web, correo electrónico, administración remota y transferencia de archivos confiar. Otras aplicaciones que no requieren de un servicio fiable de flujo de datos, puede usar el User Datagram Protocol (UDP), que proporciona un datagrama servicio que hace hincapié en reducir la latencia de más de fiabilidad

2.6.12 Protocolo de Internet (IP)

Es una etiqueta numérica asignada a cada dispositivo (por ejemplo, ordenador, impresora) que participan en una red informática que utiliza el protocolo de Internet para la comunicación. Una dirección IP tiene dos funciones principales: host o una red interfaz de identificación y ubicación frente. Su papel se ha caracterizado de la siguiente manera: " A nombre de... indica lo que buscamos una dirección donde se indica una ruta indica cómo llegar hasta allí"

Los diseñadores del protocolo de Internet definen una dirección IP como de 32-bit número, y este sistema, conocido como Protocolo de Internet versión 4 (IPv4), está todavía en uso hoy en día. Sin embargo, debido al enorme crecimiento de la Internet y la predicción de agotamiento de las direcciones disponibles , un nuevo sistema de direccionamiento (IPv6), de 128 bits para la dirección, fue desarrollado en 1995, estandarizados como RFC 2460 en 1998, y se está implementando en todo el mundo desde mediados de la década de 2000

Las direcciones IP son números binarios, pero normalmente se almacenan en archivos de texto y se muestra en legible anotaciones, como 172.16.254.1 (para IPv4), y 2001: db8: 0:1234:0:567:8:1 (por IPv6). La Internet Assigned Numbers Authority (IANA) gestiona la asignación de direcciones IP a nivel mundial del espacio y de los delegados cinco registros regionales de Internet (RIR) para asignar bloques de direcciones IP a los registros locales de Internet (Internet Service Provider) y otras entidades

2.6.13 Cliente y Servidor

Describe la relación de los programas de cooperación en la aplicación. El componente de servidor dispone de una función o servicio a uno o varios clientes, que inician solicitudes de tales servicios

Con funciones tales como el intercambio de correo electrónico, acceso a Internet y acceso a bases de datos, se basan en el modelo cliente-servidor. Usuarios que acceden a los servicios bancarios desde su ordenador utiliza un cliente de navegador web para enviar una solicitud a un servidor web en un banco. Ese programa puede a su vez remitirá la solicitud a su propio programa de cliente de base de datos que envía una petición a un servidor de base de datos en otro equipo del banco para recuperar la información de la cuenta. El saldo se devuelve al cliente de base de datos del banco, que a su vez sirve de nuevo al cliente del explorador Web que muestra los resultados al usuario. El modelo cliente-servidor se ha convertido en una de las ideas centrales de la computación en red. Muchas aplicaciones de negocios que se escriben hoy en día utilizan el modelo cliente-servidor. Lo mismo ocurre con los protocolos principales de Internet de aplicaciones, tales como HTTP, SMTP, Telnet y DNS

La interacción entre el cliente y el servidor se describe a menudo usando diagramas de secuencia. Los diagramas de secuencia se han estandarizado en el Lenguaje de Modelado Unificado

Los tipos específicos de clientes se incluyen navegadores web, clientes de correo electrónico y chat en línea los clientes. Los tipos específicos de servidores incluyen los servidores web, servidores FTP, servidores de aplicaciones, servidores de bases de datos, servidores de nombres, servidores de correo, servidores de archivos, servidores de impresión y servidores de terminales

2.6.14 Global System Mobile (GSM)

Es un conjunto estándar desarrollado por el European Telecommunications Standards Institute (ETSI) para describir las tecnologías de segunda generación (o " 2G ") digital de las redes celulares. Desarrollado como un reemplazo de la primera generación de redes celulares analógicas, el estándar GSM descrito originalmente una cámara digital, conmutación de circuitos de la red optimizada para full duplex de voz de telefonía. La norma se amplió el tiempo para incluir circuito enciende por primera vez el transporte de datos, de transporte de paquetes de datos a través de GPRS. Velocidades de transmisión de paquetes de datos se incrementaron más tarde a través de EDGE. El estándar GSM es seguido por la tercera generación (o"3G") UMTS estándar desarrollado por el 3GPP. Redes GSM se desarrollará aún más a medida que comienzan a incorporar la cuarta generación (o " 4G ") LTE Advanced normas. "GSM" es una marca comercial propiedad de la Asociación GSM [18]. La GSM Association estima que las tecnologías definidas en la norma GSM servir el 80% de la población mundial, que abarca más de 5 millones de personas en más de 212 países y territorios, siendo GSM la más ubicua de las numerosas normas para redes celulares

2.6.15 Global Position System (GPS)

Es un espacio basado en sistema mundial de navegación por satélite (GNSS) que permite la ubicación de la información y el tiempo en todo tiempo y lugar en o cerca de la Tierra, donde hay una línea de visión sin obstáculos a cuatro o más satélites GPS. Es mantenida por la de los Estados Unidos el gobierno y es de libre acceso por cualquier persona con un receptor GPS con algunas limitaciones técnicas que sólo se eliminan para los usuarios militares

El proyecto de GPS fue desarrollado en 1973 para superar las limitaciones de los anteriores sistemas de navegación, la integración de las ideas de varios predecesores, incluyendo una serie de clasificar los estudios de ingeniería de

diseño de la década de 1960. GPS fue creado y realizado por el Departamento de Defensa de EE.UU. (USDOD) y se ejecutó inicialmente con 24 satélites. Que entró en pleno funcionamiento en 1994

Además de los GPS, los sistemas de otros están en uso o en desarrollo. El ruso Sistema Global de Navegación por Satélite (GLONASS) se encontraba en su uso por el ejército ruso, hasta que fue totalmente a disposición de los civiles en 2007. También hay prevista la china Brújula sistema de navegación y de la Unión Europea el sistema Galileo de posicionamiento

2.6.16 SMS

2.6.16.1 Servicio SMS

El servicio SMS, esquematizado en la Figura 4.3, permite transferir un mensaje de texto entre una estación móvil (MS) y otra entidad (SME) a través de un centro de servicio (SC). El servicio final ofrecido es una comunicación extremo-extremo entre la estación móvil (MS) y la entidad (SME). La entidad puede ser otra estación móvil o puede estar situado en una red fija. En el caso de envío de un mensaje entre dos móviles, ambas partes son estaciones móviles. Cuando se envía un mensaje para solicitar algún tipo de servicio de valor añadido, un extremo es una estación móvil y la otra es un servidor que atiende las peticiones, como puede ser uno de los exitosos sistemas de televoto actuales. En la norma GSM sólo se especifica la parte de comunicaciones entre las Estaciones móviles (MS) y el Centro de servicio. La comunicación entre el Centro de Servicio y las entidades fijas, queda fuera del ámbito de esta norma.

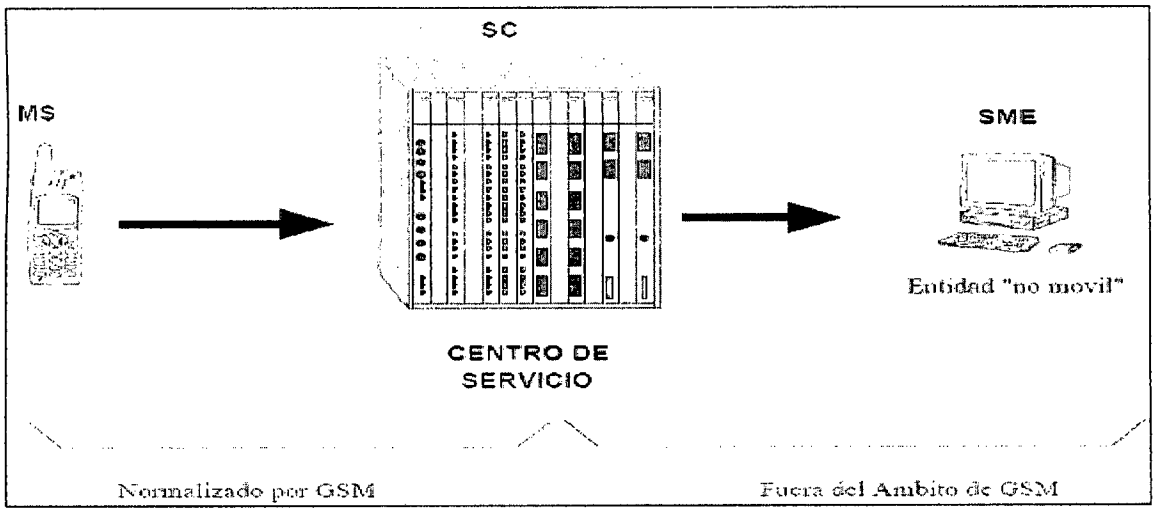


Figura 2.13 ***Estructura del servicio SMS***

El servicio SMS se divide en dos servicios Básicos detallados en la Figura 2.14

1. SM MT (Short Message Mobile Terminated Point-to-Point). Servicio de entrega de un mensaje desde el SC hasta una MS, obteniéndose un informe sobre lo ocurrido.
2. SM MO (Short Message Mobile Originated Point-to-Point). Servicio de envío de un mensaje desde una MS hasta un SC, obteniéndose un informe sobre lo ocurrido.

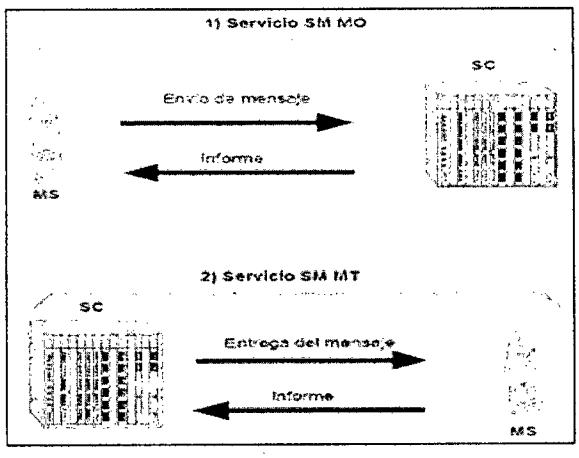


Figura 2.14 ***Servicios básicos SM MO y SM MT***

2.6.16.2 Arquitectura

La estructura básica de la red para el servicio SMS consta de las siguient es entidades:

- MS: Estación móvil.
- MSC: Centro de conmutación.
- SMS-

GMSC: MSC pasarela para el servicio de mensajes cortos (Servicio SMMT).

- SMS-IWMSC: MSC de interconexión entre PLMN y el SC (Servicio SM MO).
- SC: Centro de Servicio.
- HLR, VLR.

2.6.16.3 Modelo de capas

Para la descripción detallada de la arquitectura, se utiliza un modelo de capas (Figura 2.15), en el que cada capa o nivel proporciona un servicio a la capa superior, y este servicio se implementa mediante el protocolo correspondiente. La arquitectura se Divide en 4 capas:

- **SM-AL** (Short Message Aplication Layer): Nivel de aplicación.
- **SM-TL** (Short Message Transfer Layer): Nivel de transferencia. Servicio de Transferencia de un mensaje corto entre una MS y un SC (en ambos sentidos) y Obtención de los correspondientes informes sobre el resultado de la transmisión. Este servicio hace abstracción de los detalles internos de la red, permitiendo que el nivel de aplicación pueda intercambiar mensajes.
- **SM-RL** (Short Message Relay Layer): Nivel de repetición. Proporciona un servicio al nivel de transferencia que le permite enviar TPDU (Transfer Protocol Data Units) a su entidad gemela.
- **SM-LL** (Short Message Lower Layers): Niveles inferiores.

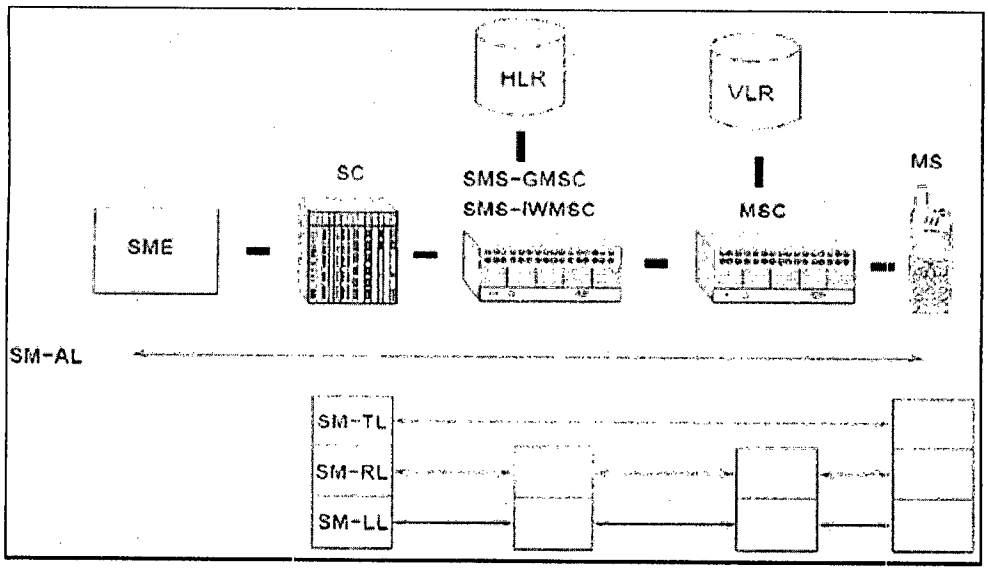


Figura 2.15 Arquitectura SMS

Cada capa proporciona los servicios a la capa superior utilizando un protocolo.

Se definen los protocolos SM-TP y SM-RP, que se corresponden con las capas SM-RL y SM-TL. El nivel de interés es el SM-TL, que es el que se usará para enviar y recibir SMS.

El servicio proporcionado por la capa SM-TL permite al nivel de aplicación enviar mensajes a su entidad gemela, recibir mensajes de ella así como obtener informes Sobre el estado de transmisiones anteriores. Para hacerlo se utilizan las siguientes PDUs:

- **SMS-DELIVER**: Transmitir un mensaje desde el SC al MS.
- **SMS-DELIVER-REPORT**: Error en la entrega (si lo ha habido).
- **SMS-SUBMIT**: Trasmitir un mensaje corto desde el MS al SC.
- **SMS-SUBMIT-REPORT**: Error en la transmisión (Si lo ha habido).
- **SMS-STATUS-REPORT**: Transmitir un informe de estado desde el SC al MS.
- **SMS-COMMAND**: Transmitir un comando desde el MS al SC.

2.6.16.4 SMS-SUBMIT

La estructura de la PDU SMS-SUBMIT se muestra en la Figura 4.6. Para el Caso también interesante de una PDU SMS-DELIVER, la estructura es tremendamente Similar y no se detallará. Los campos que la componen son los siguientes:

- **SCA**: Número de teléfono del Centro de Servicio (SC). La estructura detallada Se muestra en la Figura 2.17. Consta de los siguientes campos:

    - **Longitud**: Número de dígitos del teléfono del SC.
    - **Tipo de número**: Indica si se trata de un número nacional o internacional:

        81h: Nacional.

        91h: Internacional.
    - **Dígitos BCD**: Número de teléfono del SC, en dígitos BCD.

- **PDU-TYPE**: Contiene información sobre el tipo de PDU:
    - **RP**: Existe camino de respuesta. RP=0 en tramas de tipo SMS-SUBMIT.
    - **UDHI**: Indica si el campo UD contiene sólo el mensaje corto (UDHI=0) o si existe una cabecera antes del mensaje corto (UDHI=1).
    - **SRR**: Informe de estado no solicitado (SRR=0) o sí solicitado (SRR=1).
    - **VPF**: Indica si el campo VP está o no presente.
    - **RD**: Rechazar o no duplicados.
    - **MTI**: Tipo de mensaje.
- **MR**: Parámetro para identificar el mensaje.
- **DA**: Dirección del SME destino (número de teléfono).
- **PID**: Identificación del protocolo de la capa superior.
- **DCS**: Identificación del tipo de codificación dentro de los datos de usuario.
- **VP**: Periodo de validez del mensaje.
- **UDL**: Longitud del campo UD.

- **UD**: Datos de usuario.

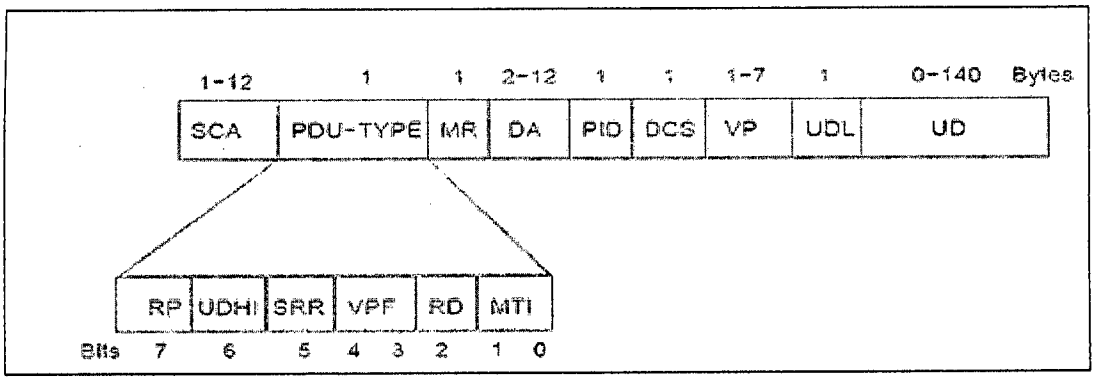


Figura 2.16 ***Estructura de la PDU SMS-SUBMIT***

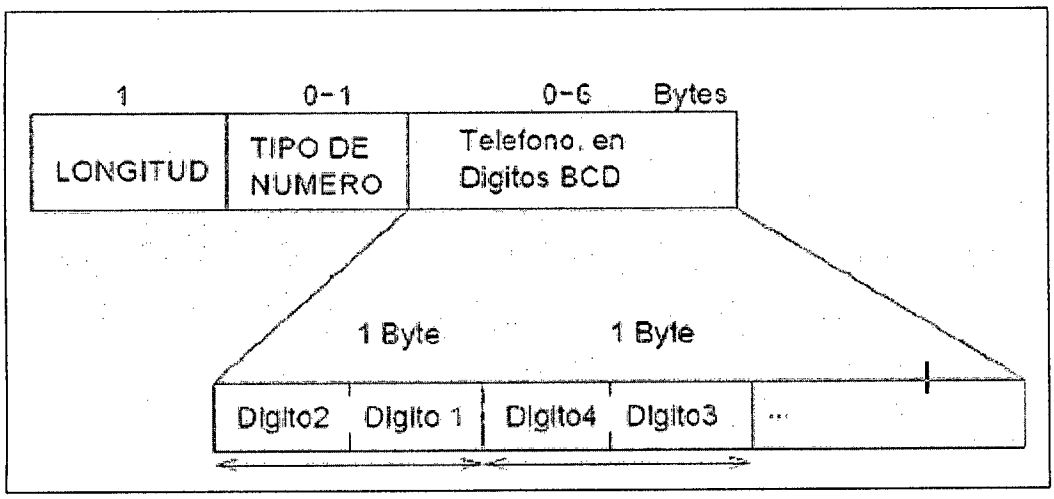


Figura 2.17 ***Detalle del campo SCA***

Si quisiéramos enviar el mensaje corto "hola" al teléfono 630672901 utilizando el Centro de mensajes +341710760000.

SCA: 0C91437101670000 (8 bytes)

Longitud	Tipo	Tlf en BCD
0C	91	43-71-01-67-00-00

PDU-TYPE: 01h. Trama de tipo **SMS-SUBMIT**. Campo de usuario sin cabecera. Informe de estado no solicitado. Campo VP no presente.

RP	UDHI	SRR	VPF	RD	MTI
0	0	0	0 0	0	0 1

MR: 00h. Número de referencia 0.

DA: 0681366027091F (7 bytes). Teléfono destino.

Longitud	Tipo	Tlf en BCD
09	81	36-60-27-09-F1

PID: 00h (mensaje corto).

- **DCS**: F6h (Codificación de 8 bits, en ASCII).
- **UDL**: 04. Longitud de los datos de usuario.
- **UD**: 686F6C61 (4 bytes). Datos de usuario.

h	o	l	a
68	6F	6C	61

La trama final montada es la mostrada en la Figura 2.18 que ocupa un total de 24 bytes para enviar sólo cuatro.

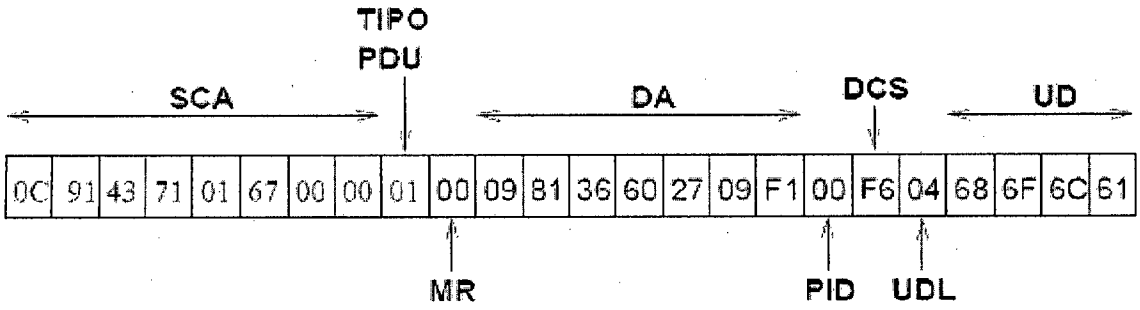


Figura 2.18 Ejemplo de PDU SMS

2.6.16.5 Los comandos AT

Los comandos AT (se denominan así por la abreviatura de *attention*) son instrucciones Codificadas que conforman un lenguaje de comunicación entre el hombre y un terminal módem. En un principio, el juego de comandos AT fue desarrollado en 1977 por Dennis Hayes como un interfaz de comunicación con un módem para así poder configurarlo y Proporcionarle instrucciones, tales como marcar un número de teléfono. Más adelante, Con el vance del baudio, fueron las compañías Microcomm y US Roboticsl las que Siguieron desarrollando y expandiendo el juego De comandos hasta universalizarlo. Aunque la finalidad principal de los comandos AT es la comunicación con módems, la telefonía móvil GSM también ha adoptado como estándar este lenguaje para poder comunicarse con sus terminales. De esta forma, todos los teléfonos móviles GSM poseen un juego de comandos AT específico que sirve de interfaz para configurar y proporcionar instrucciones a los terminales. Este juego de instrucciones puede encontrarse en la documentación técnica de los terminales GSM y permite acciones tales como realizar llamadas de datos o de voz, leer y escribir en la

agenda de contactos y enviar mensajes SMS, además de muchas otras opciones de configuración del terminal.

Los comandos AT con cadenas ASCII que comienzan por los caracteres AT y terminan con un *retorno de carro (LF)*. Cada vez que el módem recibe un comando, lo Procesa y devuelve un resultado, que normalmente es una cadena ASCII salvo que hayamos indicado lo contrario. Al estar la comunicación en ASCII, pondremos utilizar Un terminal de comunicaciones desde un ordenador para acceder al módem, bien para configurarlo, bien para hacer pruebas o bien para establecer una comunicación con otro módem.

Los módems GSM no sólo se comportan de forma muy parecida a un módem normal, permitiendo el intercambio de datos con otro módem y utilizándose los comandos AT originales, sino que incluyen muchas más características.Son como pequeños teléfonos móviles, que incluyen su propia tarjeta SIM para poder funcionar y por tanto permiten gestionar la base de datos de teléfonos, la lista de los mensajes SMS recibidos, enviar mensajes SMS, configurar diversos parámetros. Para tener acceso a todos esos servicios, y dado que los comandos AT estaban muy extendidos y muy estandarizados, se ha realizado una ampliación, añadiéndose nuevos comandos. Estos nuevos comandos comienzan por los caracteres AT+, y se denominan comandos AT+. Mostramos a continuación la estructura general de estos comandos en un sencillo ejemplo:

- Petición:

  AT+CMGI<CR> *//Donde <CR> simboliza el retorno de carro.*

- Respuesta correcta:

  <CR><LF>Siemens mobile phones<CR><LF>

  <CR><LF>OK<CR><LF> *//Donde <LF> simboliza nueva línea.*

- Respuesta errónea:

  <CR><LF>ERROR<CR><LF>

Indiquemos, llegados a este punto, que el código no es sensible al uso de mayúsculas o minúsculas y que la tecla "intro" de un PC tipo qwerty envía al terminal los caracteres retorno de carro y nueva línea.

2.6.16.6 Listado de comandos AT y AT+ más frecuentes

1. **Comandos generales**
    a. **AT+CGMI**: Identificación del fabricante.
    b. **AT+CGSN**: Obtener número de serie.
    c. **AT+CIMI**: Obtener el IMSI.
    d. **AT+CPAS**: Leer estado del módem.

2. **Comandos del servicio de red**
    a. **AT+CSQ**: Obtener calidad de la señal.
    b. **AT+COPS**: Selección de un operador.
    c. **AT+CREG**: Registrarse en una red.
    d. **AT+WOPN**: Leer nombre del operador.

3. **Comandos de seguridad:**
    a. **AT+CPIN**: Introducir el PIN.
    b. **AT+CPINC**: Obtener el número de reintentos que quedan.
    c. **AT+CPWD**: Cambiar password.

4. **Comandos para la agenda de teléfonos**
    a. **AT+CPBR**: Leer todas las entradas.
    b. **AT+CPBF**: Encontrar una entrada.
    c. **AT+CPBW**: Almacenar una entrada.
    d. **AT+CPBS**: Buscar una entrada.

5. **Comandos para SMS**
    a. **AT+CPMS**: Seleccionar lugar de almacenamiento de los SMS.
    b. **AT+CMGF**: Seleccionar formato de los mensajes SMS.
        i. Modo texto
        ii. Modo PDU
    c. **AT+CMGR**: Leer un mensaje SMS almacenado.
    d. **AT+CMGL**: Listar los mensajes almacenados.
    e. **AT+CMGS**: Enviar mensaje SMS.
    f. **AT+CMGW**: Almacenar mensaje en memoria.
    g. **AT+CMSS**: Enviar mensaje almacenado.

h. **AT+CSCA**: Establecer el Centro de mensajes a usar.

i. **AT+ WMSC**: Modificar el estado de un mensaje.

2.6.16.7 Algunos ejemplos

A continuación se muestran algunos ejemplos de utilización de los comandos AT+. Para probarlos se ha utilizado un ordenador PC, un módem GSM conectado al puerto serie y un terminal de comunicaciones. En nuestro caso, se ha empleado la aplicación "Hyperterminal" de Microsoft que se incluye como utilidad en cualquier sistema operativo Windows.

Listado de mensajes

Los mensajes cortos se dividen en 5 categorías, cada una identificada por una cadena. Para listar los mensajes se utiliza el comando **AT+CMGL=<categoría>**, donde *<categoría>* es una cadena de texto que puede valer lo siguiente:

- "**REC UNREAD**": Mensajes recibidos pero no leídos.
- "**REC READ**": Mensajes recibidos y leídos.
- "**STO UNSEND**": Mensajes escritos y almacenados pero no enviados.
- "**STO SENT**": Mensajes enviados.
- "**ALL**": Todos los mensajes.

A continuación se leen todos los mensajes:

AT+CMGL="ALL"

+CMGL: 1,"REC READ","609",,"05/02/27,18:16:51+40"

Como cliente MoviStar Plus Elección, esta de enhorabuena. Porque desde el 18 de febrero esta ahorrando un 49 % en sus llamadas de móvil a fijo en horario normal

+CMGL: 2,"REC READ","1122",,"05/02/28,20:41:25+40"

-Bienvenido a Omitel Movistar! Para acceder a su buzón de voz marque 123, servicio de Atención al Cliente marque 609 (llamadas no gratuitas desde el extranjero)

+CMGL: 3,"REC READ","+34609100609",,"05/05/06,10:00:16+04"

Telefónica MoviStar le desea una feliz estancia. Para llamar al CRC MoviStar marque +34 609 100 609. Para llamar A su Buzón de Voz marque +34 609 123 123
OK

Lectura de un mensaje

Se utiliza el comando **AT+CMGR=<número>**, donde *<número>* es el número del mensaje a leer.

AT+CMGR=1

+CMGR: "REC READ","609",,"05/02/27,18:16:51+40"
Como cliente MoviStar Plus Elección, esta de enhorabuena.
Porque desde el 18 de febrero esta ahorrando un 49 % en sus llamadas de móvil a fijo en horario normal
OK

Si se especifica un número de mensaje que no existe se devuelve un mensaje de error:

AT+CMGR=4
ERROR

Borrar un mensaje

Se utiliza el comando **AT+CMGD=<numero>**,donde <número>hace Referencia al número de mensaje a borrar.

AT+CMGD=3
OK

Mensaje Borrado. Si ahora se intenta leer:

AT+CMGR=3
ERROR

Envío de un SMS en modo texto

Para enviar un mensaje SMS se puede realizar de dos maneras diferentes. Se puede utiliza el **modo texto**, en que sólo hay que indicar el número de teléfono y el Contenido del mensaje. Es el módem el que se encarga de generar la trama SMS SUBMIT correspondiente y enviarla. Este es el modo que normalmente se emplea si sólo queremos transmitir un mensaje pues simplifica mucho el proceso.

Es posible tener acceso directamente al protocolo **SM-TP**, enviando Directamente una trama de tipo SMS-SUBMIT. En este caso se habla de **modo PDU**.
Será el nivel de aplicación el que tendrá que generar correctamente la trama SMS-SUBMIT y el módem simplemente la transmitirá.
La configuración del módem para funcionar en uno u otro modo se realiza Mediante el comando **AT+CMGF=<modo>**, donde <*modo*>puede tener los siguientes valores:

- <*modo*>=1: **Modo texto**
- <*modo*>=0: **Modo PDU** (Modo por defecto)

Para enviar un mensaje en **modo texto**, se utiliza el comando **AT+CMGS**.
Primero se especifica el número de teléfono, seguido de un carácter retorno carro <CR>

El módem responde enviando el carácter ">" que indica que se puede escribir el mensaje que se quiere enviar. Para delimitar el mensaje hay que enviar el carácter

<*control-z*> (Es el carácter ASCII 26).
Si el mensaje se ha enviado correctamente, devuelve la cadena "+CMGS:<nr>" seguida del OK. El campo <*nr*> es el número de referencia del mensaje, que se va incrementando, tomando los valores comprendidos entre 0 y 255, cada vez que se envía un SMS.

AT+CMGS="630672901"<CR>
>Mensaje de prueba <control-z>
+CMGS: 2
OK
Puesto que hemos enviado un auto-mensaje (un mensaje SMS con destino el mismo móvil que lo ha originado), al cabo de un cierto tiempo se recibe el mensaje, por lo que aparece en el terminal lo siguiente:

+CMTI: "SM",3

Que indica que se ha recibido un mensaje SMS y se ha almacenado con el número 3. Si ahora leemos el mensaje:

AT+CMGR=3

+CMGR: "REC UNREAD","+34630672901",,"05/06/23,11:57:20+00"

Mensaje de prueba

OK

La información que se obtiene es la siguiente. Primero el estado del mensaje, "REC UNREAD", para indicar que es un mensaje nuevo que no se había leído. A Continuación el teléfono del remitente, la fecha y la hora en la que se ha recibido y Finalmente el mensaje recibido. Si ahora se vuelve a leer el mensaje, el estado será "REC READ". En caso de no haber cobertura a la hora de enviar el mensaje, el Comando AT+CMGS devuelve la cadena ERROR.

AT+CMGS="630672901"<CR>

>Mensaje de prueba <control-z>

ERROR

2.6.16.8 Módem GSM

2.6.16.8.1 Introducción

Actualmente están apareciendo gran cantidad de servicios basados en mensajes cortos. Además de ser usados para enviar mensajes de texto entre personas, de forma Simular a los busca ,se están ofreciendo otros servicios como son:

- Votaciones mediante SMS.
- Suscripción a servicios de información.
- Informe de averías en ciertos equipos. Por ejemplo, muchos cajeros automáticos envían un SMS al servicio técnico cuando detectan que hay alguna avería o les falta algún recurso: dinero, papel...
- Ofrecer servicios de soporte a otras empresas. Como la empresa Pulsar Technologies, que ofrece soporte con las impresoras de HP.

Para poder ofrecer estos servicios es necesario diseñar software y hardware que pueda acceder a los servicios SMS. Esto se puede conseguir de varias maneras:

1. Algunos teléfonos se pueden conectar directamente a un PC y mediante un Software propietario se puede acceder a los datos de móvil (agenda, tarjeta

SIM...), así como enviar y recibir mensajes SMS. El principal problema de esta Solución es que no es abierta, y los fabricantes no proporcionan suficiente Información como para poder realizar aplicaciones con ellos, siendo necesario Realizar ingeniería inversa. No obstante, hemos de incidir en este punto que dada La profusión que esta alcanzando el mercado móvil en general y en nuestro país en particular, los fabricantes están intentando diversificar su mercado y es obvio que el mundo del hardware para telecontrol no les es ajeno. Es por esto que poco a poco la utilización de comandos AT y AT+ estándar se esta generalizando. No obstante, aún existen diferencias entre los terminales modernos de la llamada segunda generación, aunque atenuada en la llamada generación 2.5 (GSM+GPRS), sobre todo por la posibilidad que incorporan los terminales de poder ser utilizados como módem. Otro problema remanente a esta solución es el interfaz hardware para estos terminales, pues no siempre es fácil encontrar en el mercado conectores adecuados para lograr el diálogo M2M (Machine-to- Machine). A pesar de todo lo indicado, se realizaron una serie de pruebas con un terminal Siemens S55 conectado al puerto serie de un PC y fue posible configurarlo y utilizarlo del mismo modo que el módem utilizado.

2. Utilización de un módem GSM, (es la solución adoptada). Mediante un módem GSM podemos conectar cualquier sistema digital a la red GSM, no sólo para enviar mensajes SMS sino también para transmitir datos. Existen dos tipos de módems, según la aplicación que queramos realizar.

   a. módems para circuito impreso: Son módems de reducido tamaño (como una tarjeta de memoria aprox.) y perfectamente apantallados que están preparados para ser incorporados dentro de un circuito impreso y que permiten desarrollar un hardware específico y que no depende de un PC.

   b. módems para PC. Fue la elección final, tienen un tamaño también bastante reducido, y disponen de un conector DB9 hembra para conectarse al PC a través de un cable serial. Son útiles para que desde cualquier ordenador de una intranet se puedan enviar mensajes SMS.

2.7 SISTEMA OPERATIVO ANDROID

2.7.1 ¿QUÉ ES ANDROID?

Android es un sistema operativo para dispositivos móviles como teléfonos inteligentes y tabletas basado en el núcleo Linux. Es desarrollado por la Open Handset Alliance, la cual es liderada por Google, usando diversos conjuntos de herramientas de software de código abierto para dispositivos móviles. Fue construido para permitir a los desarrolladores la creación de aplicaciones móviles que aprovechan al máximo el uso de todas las herramientas que un dispositivo como este puede ofrecer.

Implementa una arquitectura en la que cualquier aplicación puede obtener acceso a las capacidades del teléfono móvil. Por ejemplo, una aplicación puede llamar una o varias de las funcionalidades básicas de los dispositivos móviles, tales como realizar llamadas, enviar mensajes de texto, o utilizar la cámara, facilitando a los desarrolladores crear experiencias más ricas y con más coherencia para los usuarios.

Está construido sobre el kernel de Linux. Además, se utiliza una máquina personalizada virtual que fue diseñada para optimizar los recursos de memoria y de hardware en un entorno móvil. Android es de código abierto, y además puede ser libremente ampliado para incorporar nuevas tecnologías de vanguardia que van surgiendo. La plataforma continuará evolucionando a medida que la comunidad de desarrolladores trabajando juntos puedan crear aplicaciones móviles innovadoras.

2.7.2 HISTORIA DE ANDROID

Fue desarrollado por Android Inc., empresa que en 2005 fue comprada por Google, aunque no fue hasta 2008 cuando se popularizó, gracias a la unión al proyecto de Open Handset Alliance, un consorcio formado por 48 empresas de desarrollo hardware, software y telecomunicaciones, que decidieron promocionar el software libre. Pero ha sido Google quien ha publicado la mayor parte del código fuente del sistema operativo, gracias al software Apache, que es una fundación que da soporte a proyectos software de código abierto.

Dado que Android está basado en el núcleo de Linux, tiene acceso a sus recursos, pudiendo gestionarlo, gracias a que se encuentra en una capa por encima del Kernel, accediendo así a recursos como los controladores de pantalla, cámara, memoria flash.

2.7.3 VERSIONES DISPONIBLES

El sistema operativo Android, al igual que los propios teléfonos móviles, ha evolucionado rápidamente, acumulando una gran cantidad de versiones, desde la 1.0 para el QWERTY HTC G1, hasta la 4.4.2 que acaba de salir al mercado.

- **CUPCAKE: ANDROID VERSIÓN 1.5**

Características: Widgets, teclado QWERTY virtual, copy & paste, captura de vídeos y poder subirlos a Youtube directamente.

- **DONUT: ANDROID VERSIÓN 1.6**

Características: Añade a la anterior la mejoría de la interfaz de la cámara, búsqueda por voz, y navegación en Google Maps.

- **ECLAIR: ANDROID VERSIÓN 2.0/2.1**

Características: Mejoras en Google Maps, salvapantallas animado, incluye zoom digital para la cámara, y un nuevo navegador de internet.

- **FROYO: ANDROID VERSIÓN 2.2**

Características: Incluye hostpot Wifi, mejora de la memoria, más veloz, Microsoft Exchange y video-llamada.

- **GINGER BREAD: ANDROID VERSION 2.3**

Características: Mejoras del consumo de batería, el soporte de vídeo online y el teclado virtual, e incluye soporte para pagos mediante NFC.

- **HONEY COMB: ANDROID VERSION 3.0/3.4**

Características: Mejoras para tablets, soporte Flash y Divx, integra Dolphin, multitarea pudiendo cambiar de aplicación dejando las demás en espera en una columna, widgets y homepage personalizable.

- **ICE CREAM SANDWICH: ANDROID VERSION 4.0**

Características: Multiplataforma (tablets, teléfonos móviles y netbooks), barras de estado, pantalla principal con soporte para 3D, widgets redimensionables, soporte usb para teclados, reconocimiento facial y controles para PS3.

2.7.4 ARQUITECTURA DE LA PLATAFORMA ANDROID

La arquitectura interna de la plataforma Android, está básicamente formada por 4 componentes: aplicaciones, armazón de aplicaciones, librerías y kernel/Linux. En la Figura 2.18, se muestran las capas que conforman el sistema operativo Android.

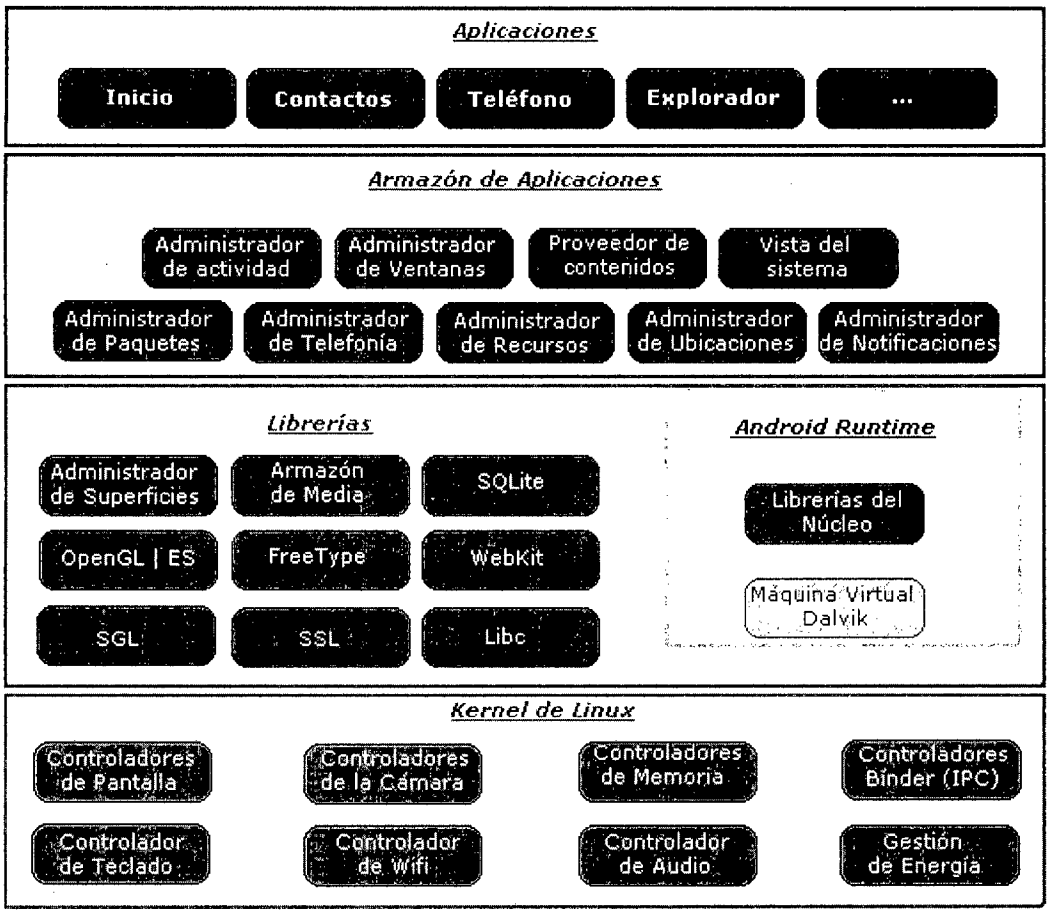


FIG. 2.18 SISTEMAS DE CAPAS DE ANDROID

2.7.5 KERNEL

Un núcleo o kernel es un software que constituye la parte más importante del sistema operativo. Es el principal responsable de facilitar a los distintos programas acceso seguro al hardware de la computadora o en forma básica, es el encargado de gestionar recursos, a través de servicios de llamada al sistema.

Como hay muchos programas y el acceso al hardware es limitado, también se encarga de decidir qué programa podrá hacer uso de un dispositivo de hardware y durante cuánto tiempo, lo que se conoce como multiplexado. Acceder al hardware directamente puede ser realmente complejo, por lo que los núcleos suelen implementar una serie de abstracciones del hardware. Esto permite esconder la complejidad, y proporciona una interfaz limpia y uniforme al hardware subyacente, lo que facilita su uso al programador.

El kernel lo podemos definir como el corazón del SO. En esta serie los modelos de desarrollo han cambiado, la manera de numerarse es de 4 dígitos (VV.RR.NR.CR).

VV: Indica la versión (o serie) del kernel.

RR: Indica la revisión del kernel (Da igual que los vea de forma impar o par, hoy en día, no tiene significado)

NR: Indica nuevas revisiones del kernel. Estos números cambian cuando se incorporan nuevas características y drivers

CR: Este digito cambia cuando se corrigen fallos de programación o fallos de seguridad dentro de una revisión.

Android utiliza el núcleo de Linux 2.6 como una capa de abstracción para el hardware disponible en los dispositivos móviles. Esta capa contiene los drivers necesarios para que cualquier componente hardware pueda ser utilizado mediante las llamadas correspondientes. Siempre que un fabricante incluye un nuevo elemento de hardware, lo primero que se debe realizar para que pueda ser utilizado desde Android es crear las librerías de control o drivers necesarios dentro de este kernel de Linux embebido en el propio Android.

2.7.6 GUIA PARA DESARROLLADORES (BÁSICA)

La Guía ofrece una introducción práctica a desarrollar aplicaciones para Android y documentación sobre las características de las plataformas más importantes. Se estudian los conceptos detrás de Android, el marco para la construcción de una aplicación y las herramientas para el desarrollo, la prueba y la publicación de software para la plataforma.

La Guía del desarrollador tiene la mayor parte de la documentación para la plataforma Android, con excepción de material de referencia en el Framework de la API, para conocer las especificaciones de la API.

ANDROID BASICS

Es una orientación inicial para Android, lo que es, lo que ofrece y cómo se ajusta su aplicación.

TEMAS DEL FRAMEWORK

Son las discusiones sobre determinadas partes del framework de Android y la API. Para una introducción al marco, comienzan con Application Fundamentals; luego explora otros temas, desde el diseño de una interfaz de usuario y la creación de recursos para el almacenamiento de datos; y el uso de los permisos.

TEMAS DEL ANDROID MARKET

La documentación de los temas que conciernen a la publicación y monetización de las aplicaciones en AndroidMarket, por ejemplo, cómo hacer cumplir las políticas de concesión de licencias y ponerlo en práctica en la aplicación de facturación.

DESARROLLO

Indicación es para el uso de desarrollo de Android y herramientas de depuración, y para comprobar los resultados.

PUBLICACIÓN

Las instrucciones sobre cómo preparar su aplicación para la implementación y la forma de publicar cuando esté listo.

APLICACIONES WEB

Hace referencia a la documentación sobre cómo crear aplicaciones web que funcionan perfectamente en dispositivos con Android y como crear aplicaciones Android para incrustar contenido basado en web.

APÉNDICE

Información de referencia y especificaciones, así como preguntas frecuentes, un glosario de términos, y otra información. El primer paso en la programación para Android es la descarga del SDK (software development kit). Después de tener el SDK, se empieza por buscar a través de the Dev Guide. Si se quiere empezar por conseguir un rápido vistazo a algo de código, el tutorial Hola Mundo los lleva a través de una aplicación "Hello World" para introducir algunos conceptos básicos de una aplicación Android.

2.7.7 PAUTAS PARA LAS INTERFACES DE USUARIO

El sistema operativo Android fue adoptado con gran rapidez por múltiples fabricantes de dispositivos móviles, que lo adoptaron como plataforma debido a su carácter más abierto.

Esto potenció a Android, y su tienda de aplicaciones (el AndroidMarket) comenzó a contar con una cantidad enorme de desarrollos (a la fecha, más de 400.000). Sin embargo, esto mismo generó en Android una falta de estandarización en las diferentes aplicaciones, que ha generado también ligeras incompatibilidades entre las distintas implementaciones del sistema operativo de los fabricantes.

Para la última versión mayor de Android (4.0) Google comenzó a tomar algunas cartas en el asunto, y para ello liberó una página llamada "AndroidDesign" que define una serie de principios para el diseño de interfaces y afirman:

"Estos principios de diseño fueron desarrollados por y para el Equipo de Experiencia de Usuario de Android teniendo en mente las mejores intenciones y consideraciones. Se deben tener en cuenta a la hora de aplicar las ideas creativas de diseño." [9] versión 4.0 de Android, pocas aplicaciones del AndroidMarket las cumplen, y en este momento no son un requerimiento para conseguir la aprobación en la tienda oficial de Android. La página de

"AndroidDesing" inicia planteando una serie de principios generales para el diseño de interfaces gráficas.

Las aplicaciones nativas de Android siguen tres principios generales:

Encántame: Las aplicaciones deben combinar belleza, simplicidad y propósito, para crear experiencias poderosas y de mínima dificultad de uso.

Simplifica mi vida: Las aplicaciones de Android deben facilitar la vida y ser fáciles de entender. Cuando las personas utilicen por primera vez una aplicación, deben deducir de manera intuitiva las características más importantes.

Sorpréndeme: No es suficiente el hacer una aplicación fácil de utilizar. Las aplicaciones de Android deben empoderar a las personas a intentar nuevas cosas y a usarlas de manera creativa.

Todos estos principios generales de diseño se traducen en una serie de elementos más concretos agrupados en tres grandes grupos: Estilo, patrones y bloques de construcción.

2.8 APP INVENTOR

App Inventor es una aplicación originalmente desarrollada por Google y mantenida ahora por el Instituto de Tecnología de Massachusetts. Permite que cualquier persona, incluyendo las no familiarizadas con la programación y SDK de Android, pueda crear aplicaciones de Software para Android. Utiliza una interfaz gráfica, muy similar al Scratch y el StarLogo, que permite a los usuarios arrastrar y soltar objetos visuales para crear una aplicación que puede ejecutarse en el sistema Android.

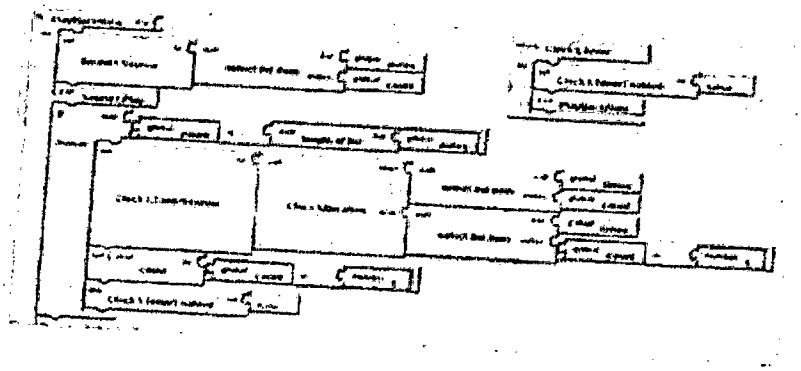


FIG. 2.19 ENTORNO DE DESARROLLO DE APP INVENTOR

Google puso fin al desarrollo el 31 de diciembre de 2011 cediéndole el código al MIT, quién lo ha puesto a disposición de todos.Se trata de una utilidad Web desarrollada por Google que permite realizar aplicaciones para Android sin escribir código Java, todo de forma visual e intuitiva (uniendo piezas de un puzle).

Una característica interesante es que el desarrollo de la aplicación es en Web. Aunque es necesario instalar un módulo de software en la computadora, en el momento del desarrollo se ejecuta la última versión del App Inventor disponible en su sitio web y los proyectos se guardan en línea. El App Inventor consta de dos segmentos principales:

- Un módulo Web y
- El editor de bloques de Android.

El módulo en Web que se mencionaba, donde aparte de ser el punto de entrada tenemos acceso a nuestros proyectos y, una vez abierto un proyecto, podemos entrar a la sección de diseño de nuestra aplicación. Esta sección es donde podemos añadir los componentes y configurarlos apropiadamente. Si se trata de componentes visuales, entonces definimos también el diseño de la interfaz. Para los familiarizados con desarrollo de aplicaciones mediante componentes visuales verán que es un concepto bastante similar.

El segmento del editor de bloques se verá más adelante, por el momento basta con mencionar que ahí es donde los bloques se conectan cual piezas Lego, formando la lógica de la aplicación.

Permite a cualquiera crear aplicaciones de software para el sistema operativo Android.

Utiliza una interfaz gráfica que permite a los usuarios arrastrar y soltar objetos visuales para crear una aplicación que puede ejecutarse en el sistema Android, que funciona en muchos dispositivos móviles.

Todo ello sin usar ni una sola línea de código, de forma intuitiva y gráfica. La aplicación se puso a disposición de los usuarios, mediante invitación, el 12 de julio de 2010, el 15 de diciembre de 2010 se puso a disposición de usuarios registrados. La aplicación está dirigida a personas que no están familiarizadas con la programación de computadoras. La idea es que cualquier persona pueda

desarrollarse sus propias aplicaciones para su dispositivo Android.

2.8.1 ¿PORQUÉ APP INVENTOR?

Porque es gratuito y poco exigente con los requisitos técnicos, funciona online y sin apenas instalación. Ha sido probado por estudiantes con excelentes resultados en institutos de enseñanza de San Francisco, eso es punto de partida interesante.

El lenguaje de programación que se usa en App Inventor es bastante similar al de Scratch, también desarrollado en el MIT. Ya hemos trabajado un poco con Scratch durante este curso. Cuando conocimos App Inventor no tuvimos ninguna duda. Había que probarlo.

2.8.2 REQUERIMIENTOS DEL SISTEMA

COMPUTADORA Y SISTEMA OPERATIVO

- **Macintosh** (with Intel processor): Mac OS X 10.5, 10.6.
- **Windows**: Windows XP, Windows Vista, Windows 7.
- **GNU/Linux**: Ubuntu 8+, Debian 5+.

NAVEGADOR WEB

- **Mozilla Firefox 3.6** o superior.
- **Apple Safari 5.0** o superior.
- **Google Chrome 4.0** o superior.
- **Microsoft Internet Explorer 7** o superior.

Para poder acceder al App Inventor se debe haber instalado Java 6 o superior en el ordenador o PC ya que no funcionará en el equipo, lo recomendable es no tratar de utilizar la aplicación App Inventor sin haberlo instalado.

Luego de ello se debe crear una cuenta en Gmail para poder tener un usuario y clave para acceder al App Inventor. La interfaz de esta herramienta se muestra a continuación:

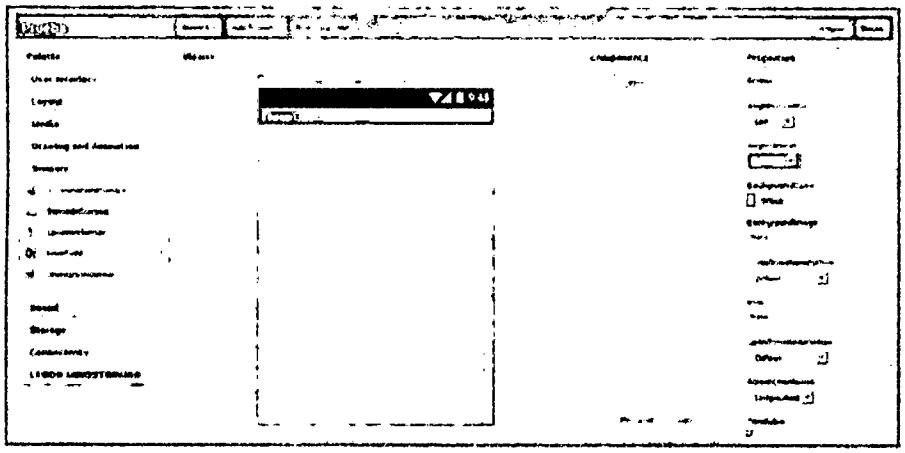
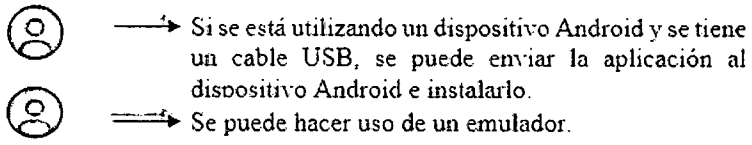


FIG. 2.20 INTERFAZ DE APP INVENTOR 2

Para realizar pruebas de la aplicación creada tenemos:



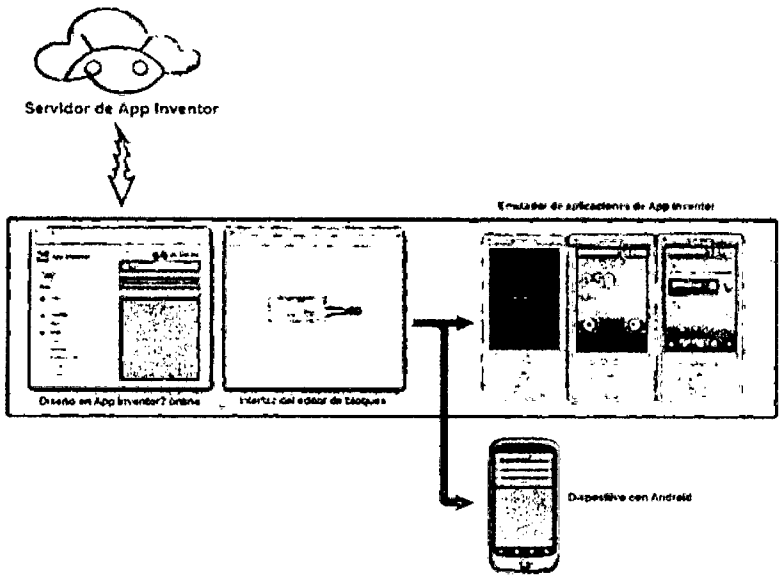


FIG. 2.21 USO DE EMULADOR VIRTUAL PARA PRUEBA DE LA APLICACIÓN

2.9 MODULACIÓN DUAL POR TONOS DE FRECUENCIA (DTMF)

En la década de los 70s la necesidad de desarrollar un método para la transferencia de la información marcado a través de la red telefónica fue reconocida. El método tradicional de señalización por marcado de pulsos, era lento, sufría varias distorsiones sobre lazos largos de cable, y además requería una ruta de corriente DC a través del canal de comunicación. Un esquema de codificación fue desarrollado utilizando tonos de frecuencia de voz, desarrollando de esta manera una alternativa muy confiable al sistema de marcado por pulsos. Este esquema de codificación es conocido como DTMF (Dual Tone Multi- Frequency), Touch-Tone™ o simplemente, marcado por tonos.

En forma resumida se puede decir que una señal DTMF válida es la suma de dos tonos, uno de un grupo de frecuencias bajas (697-941Hz) y el otro de un grupo de frecuencias altas (1209-1633Hz), donde cada grupo contiene cuatro tonos individuales. Las frecuencias de los tonos individuales fueron cuidadosamente escogidas, tal que ellos no estén relacionados armónicamente, y que el producto de su intermodulación resulte en una señal lo menos distorsionada posible, tal como se muestra en la Figura 2.22

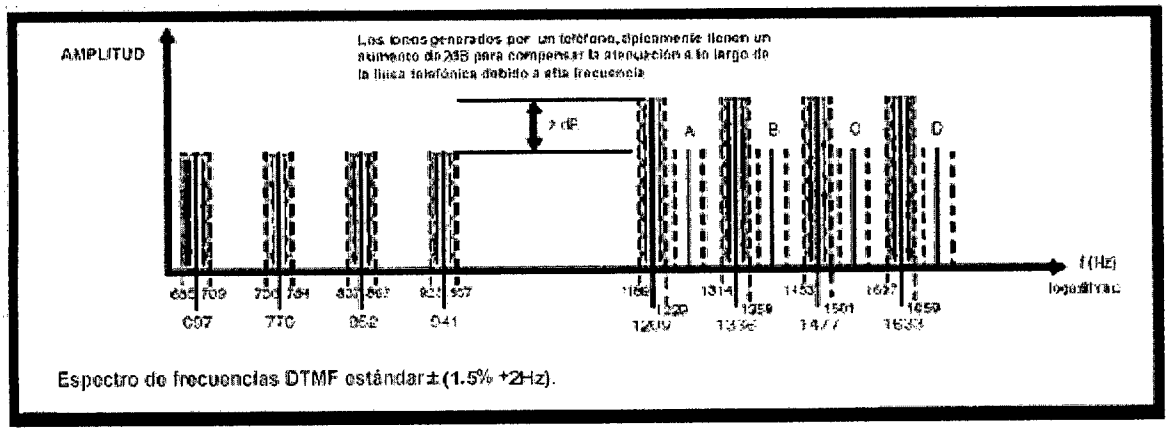


Figura 2.22 Espectro de frecuencias para los tonos DTMF

El sistema de señales DTMF es generado por un codificador, y es la suma algebraica en tiempo real de dos tonos; uno de baja frecuencia y otro de alta, el tono alto normalmente es de +1.5% (2dB) con respecto del tono bajo, para compensar pérdidas de señal en las largas líneas de conexión con la central telefónica. Este esquema permite solamente 16 combinaciones únicas. Diez de estos códigos representan los números del cero al nueve, los restantes seis (*, #, A, B, C, D) son reservados para señalización especial.

La mayoría de los teclados telefónicos contienen diez teclas numéricas, más las teclas asterisco (*) y numeral (#). Estos botones están arreglados en una matriz, que seleccionan el grupo de tonos de baja frecuencia de su respectiva fila, y el grupo de tonos de alta frecuencia desde su respectiva columna, como se muestra en la Figura 2.23.

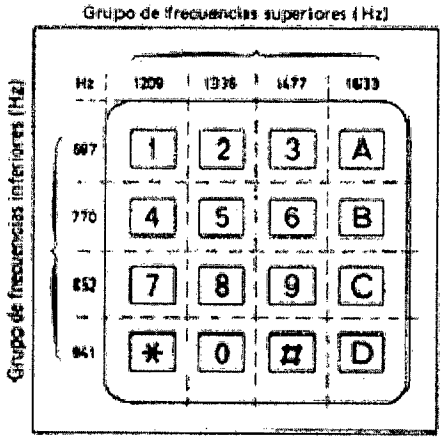


Figura 2.23 Atribución de Frecuencias a los Símbolos y Cifras del Teclado Telefónico

El esquema de codificación DTMF asegura que cada señal contiene uno y solo un componente de cada uno de los grupos de tonos alto y bajo. Esto simplifica significativamente la decodificación debido a que la señal DTMF, puede ser separada con filtros pasa banda, en sus dos frecuencias simples que la componen, cada una de las cuales puede ser manipulada en forma individual. Las teclas de función A, B, C y D son extensiones de las teclas (0-9, *, #) y fueron diseñadas con los teléfonos militares norteamericanos Autovon. Los nombres originales de estas teclas fueron FO (Flash Override), F (Flash), I (Inmediate) y P (Priority) los cuales representaban niveles de prioridad y que podían establecer comunicación telefónica con varios grados de prioridad, eliminando otras conversaciones en la red si era necesario, con la función FO siendo la de mayor prioridad hasta P la de menor prioridad. Estos tonos son más comúnmente referidos como A, B, C y D respectivamente, todos ellos tienen en común 1633 Hz como su tono alto.

2.9.1 CODIFICACIÓN DTMF

El esquema de marcado DTMF fue diseñado por los laboratorios BELL e introducido a los Estados Unidos a mediados de los años 60 como una alternativa para la marcación por pulsos, ofreciendo un incremento en la velocidad de marcado, mejorando la fiabilidad y la conveniencia de señalización de punto a punto.

Existen varias especificaciones que han sido resultado del estándar original, las cuales parten de los estándares de AT&T, CEPT, NTT, CCITT y la ITU, etc. Las variaciones de un estándar a otro son típicamente tolerancias en las desviaciones de frecuencia, niveles de energía, diferencia de atenuación entre dos tonos e inmunidad al habla. En conclusión, la codificación DTMF es el sistema de señales usado en los teléfonos para el marcado por tonos, estos son el resultado de la suma algebraica en tiempo real de dos señales sinusoidales de diferentes frecuencias. La relación de teclas con su correspondiente par de frecuencias se muestrán en la Figura 2.24.

Dígito	Frecuencia Baja	Frecuencia Alta
1	697	1209
2	697	1336
3	697	1477
4	770	1209
5	770	1336
6	770	1477
7	852	1209
8	852	1336
9	852	1477
0	941	1209
*	941	1336
#	941	1477
A	697	1633
B	770	1633
C	852	1633
D	941	1633

Figura 2.24 Pares de frecuencias empleadas para generar los tonos DTMF

2.9.2 DECODIFICACIÓN DTMF

- Las especificaciones ITU Q.23 y Q.24 para la detección DTMF son las siguientes:
- Tolerancia a la frecuencia: Un símbolo válido DTMF debe tener una desviación en frecuencia dentro del 1.5% de tolerancia. Los símbolos con una desviación en frecuencia mayor al 3.5% deberán ser rechazados.
- Duración de la señal: Un símbolo DTMF con una duración de 40ms debe ser considerado válido. La duración de la señal no debe ser menor de 23ms.
- Atenuación de la señal: El detector debe trabajar con una relación señal-ruido (SNR) de 15dB y en el peor caso con una atenuación de 26dB.
- Interrupción de la señal: Una señal DTMF válida interrumpida por 10ms o menos no debe ser detectada como dos símbolos distintos.

- Pausa en la señal: Una señal DTMF válida separada por una pausa de tiempo de al menos 40ms debe ser detectada como dos símbolos distintos.
- Fase: El detector debe operar con un máximo de 8dB en fase normal y 4dB en fase invertida.
- Rechazo al habla: El detector debe operar en la presencia del habla rechazando la voz como un símbolo DTMF válido.

Las técnicas más comunes utilizadas para el diseño de circuitos detectores de tonos DTMF se basan en el diseño de bancos de filtros de frecuencia.

CAPITULO III

DISEÑO DEL SISTEMA DE SEGURIDAD CIUDADANA USANDO LAS TECNOLOGIAS DE LA INFORMACIÓN

3.1 DISEÑO E IMPLEMENTACIÓN DE LA ALARMA COMUNIATAIRA CON TECNOLOGIA GSM

Las alarmas comunitarias son sistemas de alerta que se utilizaran en los barrios, urbanizaciones y/o zonas organizadas, con la finalidad de prevenir delitos y eventos de emergencia. Funcionan con la participación activa de los vecinos, en coordinación con la Policía y el Plan de Seguridad de la Zona o Municipio.

En la Figura 3.1 se plantea el diagrama de bloques y funcionamiento de la alarma comunitaria.

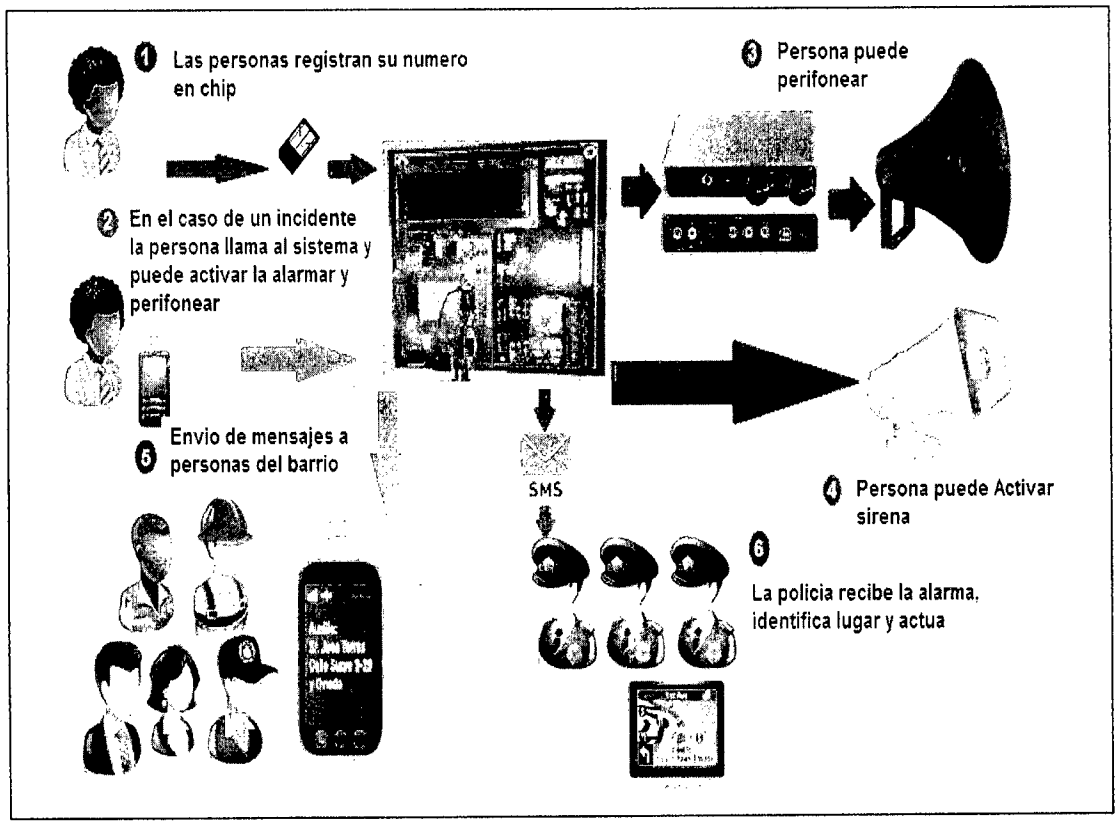


Figura 3.1 Diagrama de bloques de alarma comunitaria

Se diseñó la placa electrónica basada en el microcontrolador PIC16F877A que cumple con las características expuestas en diagrama de bloque planteado. Figura 3.2.

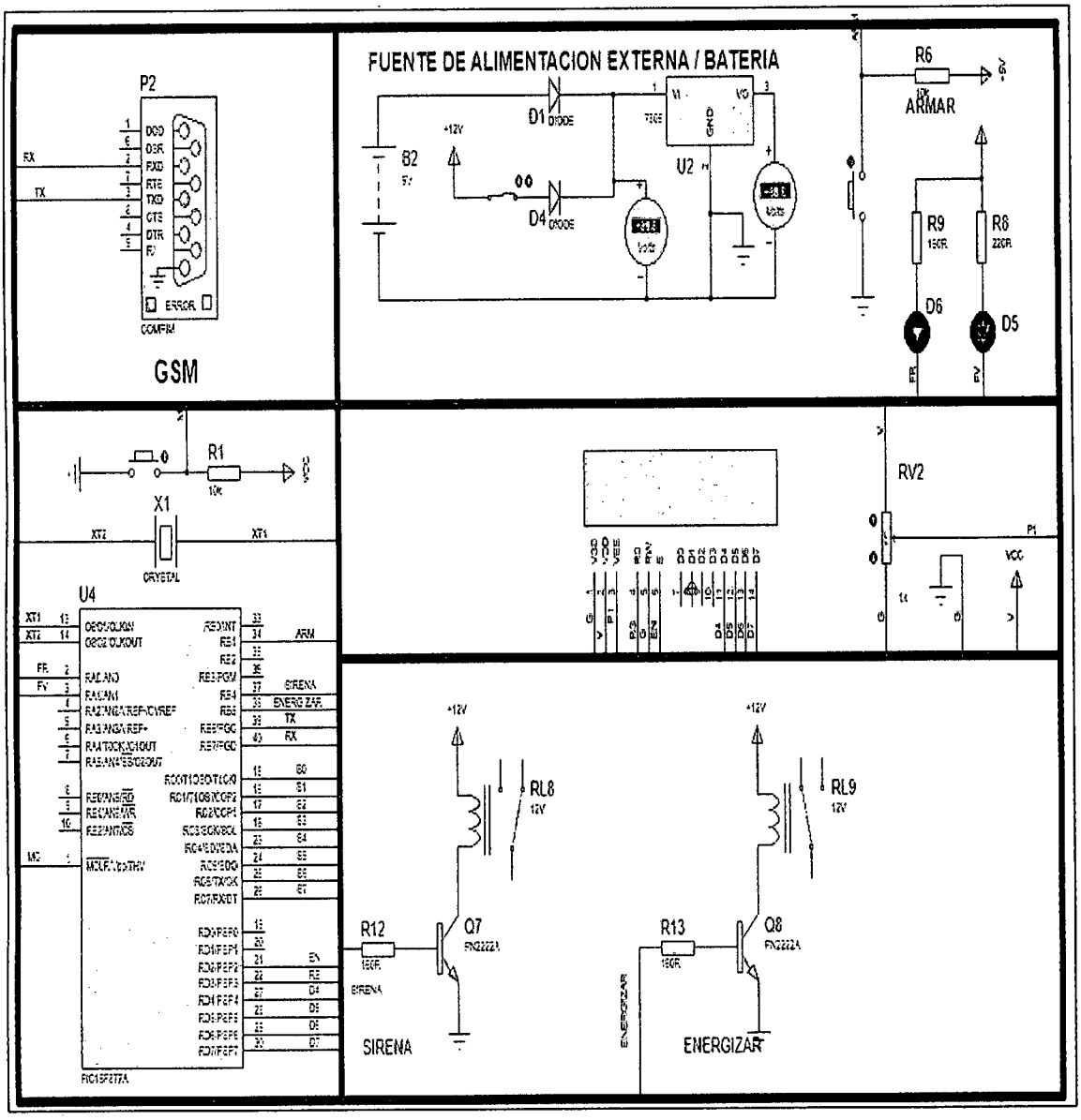


Figura 3.2 Placa Electrónica de alarma comunitaria

En la Figura 3.3 se muestra la placa en circuito impreso de la alarma comunitaria

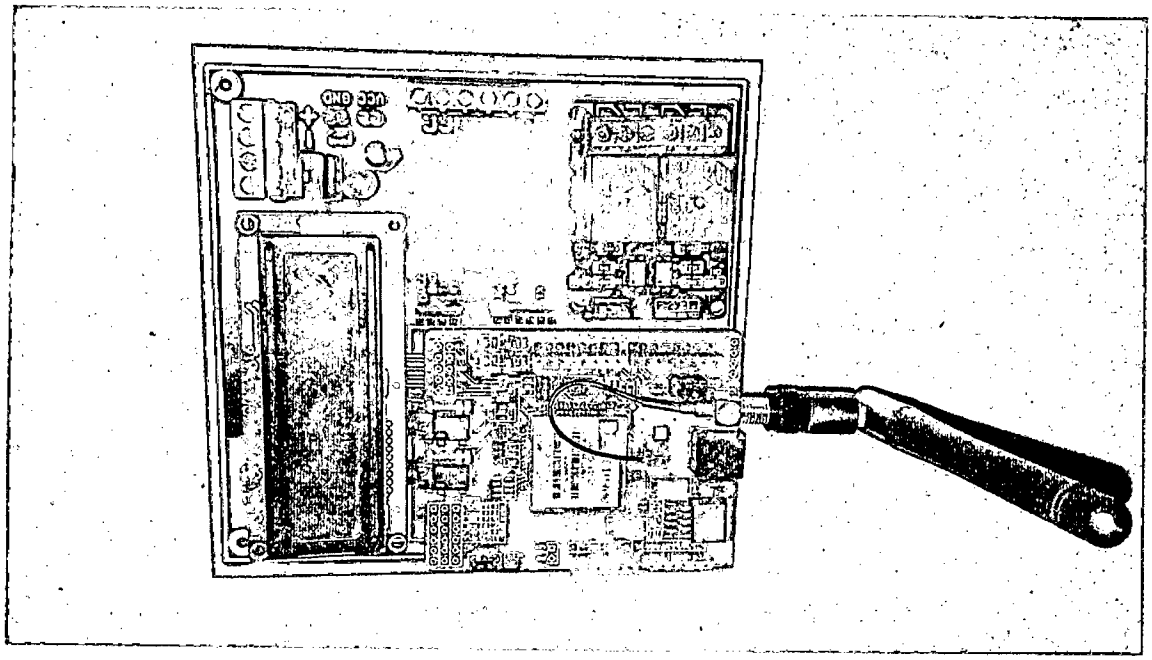


Figura 3.3 Placa de circuito impreso de alarma comunitaria

3.1.1 Microcontrolador PIC16F877A

Para la implementación de la tarjeta Electrónica se utiliza el microcontrolador PIC16F877A de Microchip que cumple con requerimientos de Puertos de E/S para el sistema que se propone así como la cantidad de memoria de Programa y Datos para el código de programa.

Los puertos de este microcontrolador se conectara como sigue: (Figura 3.4)

- Conectar 1 relé para control de sirena.
- Conectar 1 relé para para control de Bocina
- Conectar LCD de 16x2
- Comunicación RS232 con Modem GSM SIM900

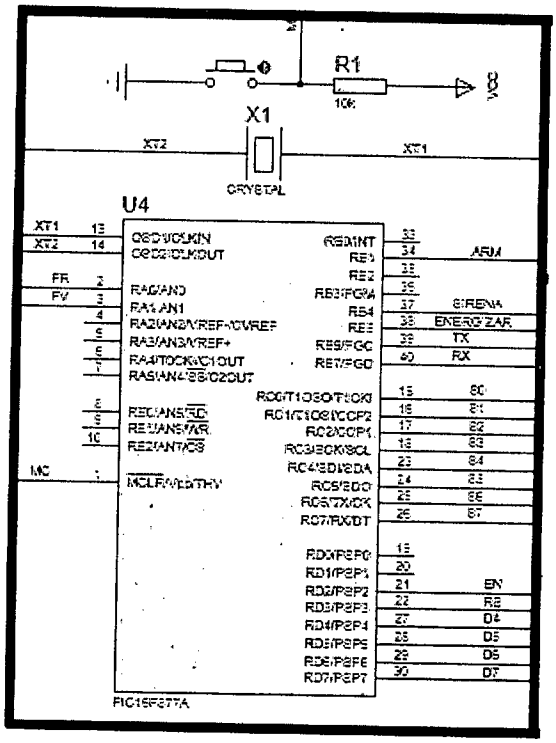


Figura 3.4 Conexiones del Microcontrolador PIC16F877A

3.1.2 Conexiones de relés

Para la conexión con relés utiliza módulos de 2 relés como el de la Figura 3.5 el cual se utiliza para conectar la sirena y la bocina, donde el circuito de la tarjeta por relé es como el de la Figura 3.6 y tiene las siguientes características:

- Voltaje de Operación 250 VAC / 30 VDC
- Voltaje de alimentación de la tarjeta del relé de 5VDC
- Corriente de operación 10 A
- Canales 2 independientes y protegidos con optoacoplador
- Indicador 1 led por cada canal

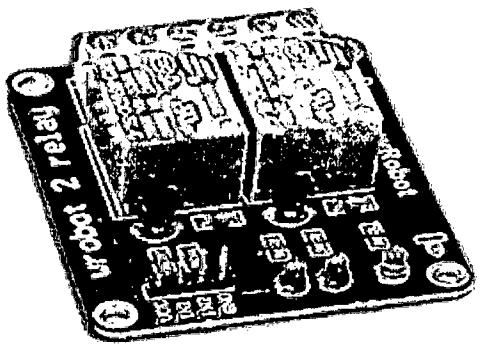


Figura 3.5 Módulo de 2 relés

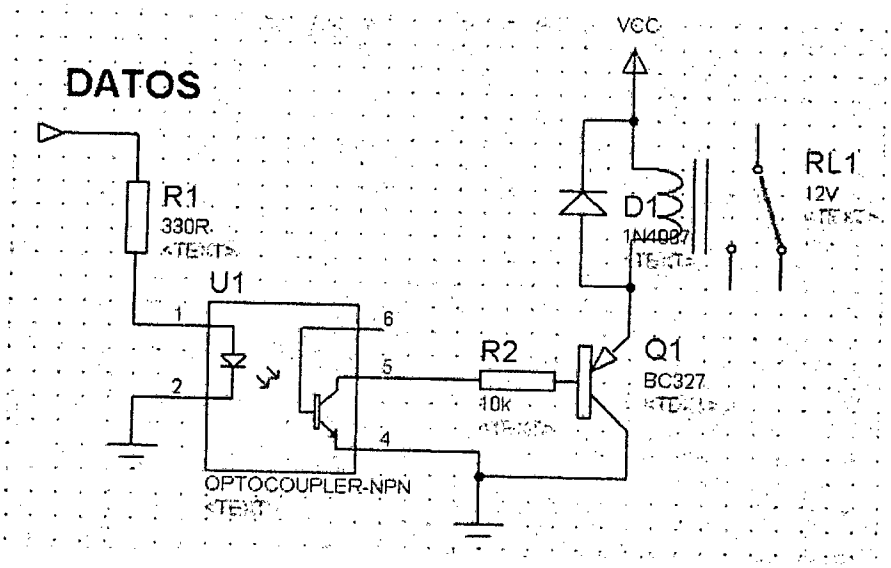


Figura 3.6 Circuito de control de relé

3.1.3 Visualización de LCD

Para nuestro sistema se utiliza un LCD de 16x2 (Figura 3.7) en el cual se visualiza los comandos recibidos, este LCD no es visible para usuarios solo para personal técnico para verificación de funcionamiento del sistema.

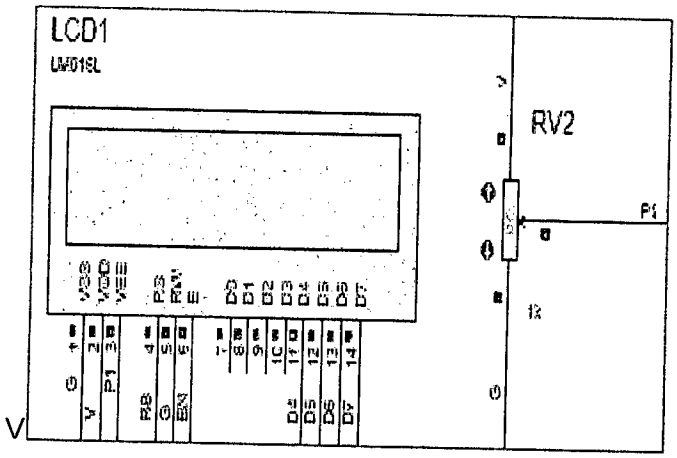


Figura 3.7 LCD 16x2

3.1.4 Fuente de Alimentación

La fuente de alimentación además de contar con una fuente de 12VDC también cuenta con una batería externa para fallos de energía eléctrica. (Figura 3.8).

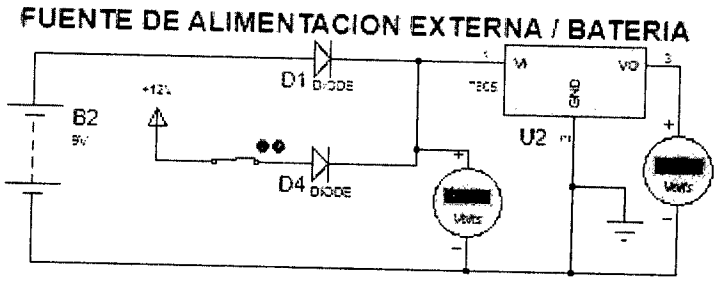


Figura 3.8 Fuente de alimentación de alarma comunitaria

3.1.5 Comunicación con el modem GSM SIM900

Para él envió de comandos por DTMF y mensajes de texto de alerta se usa el Modem GSM SIM900 (Figura 3.9).

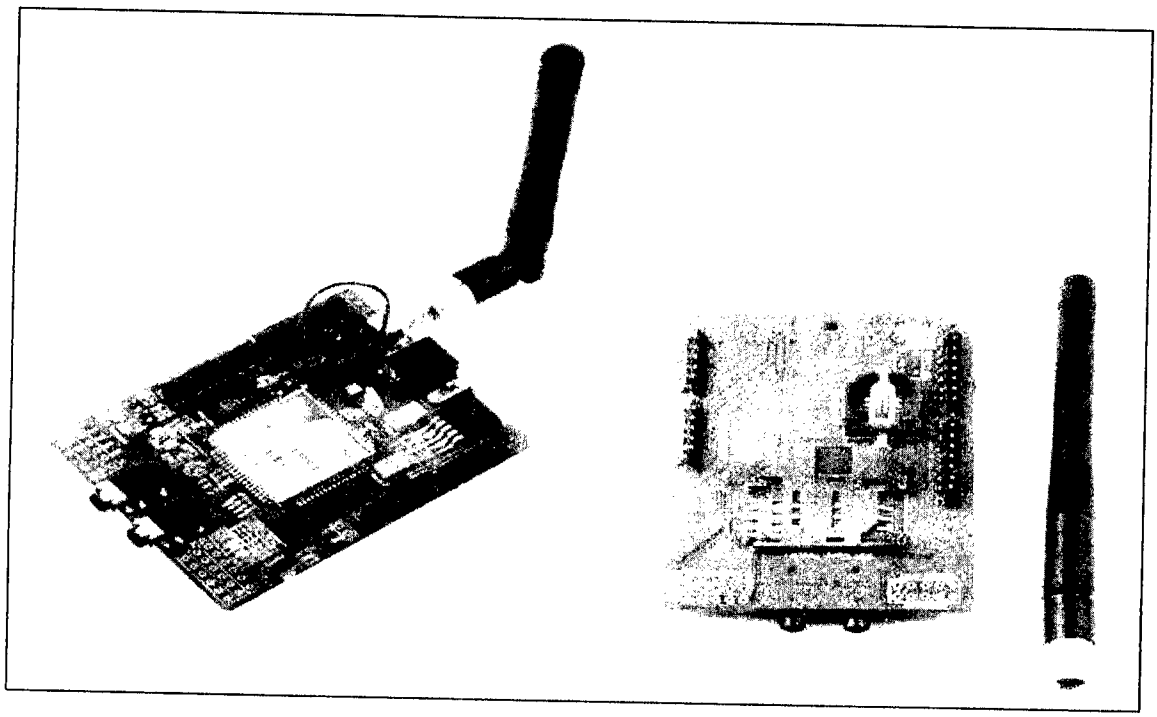


Figura 3.16 Modem GSM SIM900

El Modem GSM/GPRS se basa en el controlador SIM900 de SIMCOM y está diseñado para trabajar con microcontroladores compatibles. Permite comunicarse usando la red GSM de telefonía celular. Con este modem podrás accesar a los servicios SMS, MMS, GPRS y Telefonía de una manera sencilla enviando comandos AT. Asimismo tiene incorparadas en la placa 12 GPIOs, 2 PWM y un ADC propios del módulo SIM900.

El modem SIM900 tiene las siguientes Especificaciones:

- Quad-Band 850 / 900/ 1800 / 1900 MHz - funciona en todas las redes celulares del planeta.
- GPRS multi-slot class 10/8
- Estación móvil GPRS clase B
- Cumple con GSM phase 2/2+
- Clase 4 (2 W @ 850 / 900 MHz)
- Clase 1 (1 W @ 1800 / 1900MHz)
- Controlable vía comandos AT estándar: GSM 07.07 & 07.05 | Comandos mejorados: SIMCOM AT Commands.
- Servicio Short Message Service (SMS) - para poder enviar pequeños paquetes de datos a través de la red celular.
- Pila TCP/UDP incorporada - permite enviar datos a un servidor web. Ejem: Pachube, Fusion Tables
- Incorpora un Real Time Clock - RTC. (Requiere pila)
- Puerto serial configurable para comunicación con el microcontrolador.
- Soporte para comunicación por software Serial (pines 6 y 7).
- Jack para audífonos y micrófono (handsfree)
- Bajo consumo (en modo sleep) - 1.5mA
- Jack para alimentación externa.
- Soporta rangos temperatura - -40°C to +85 °C

En la Figura 3.17 se muestra e indican las conexiones del modem SIM900

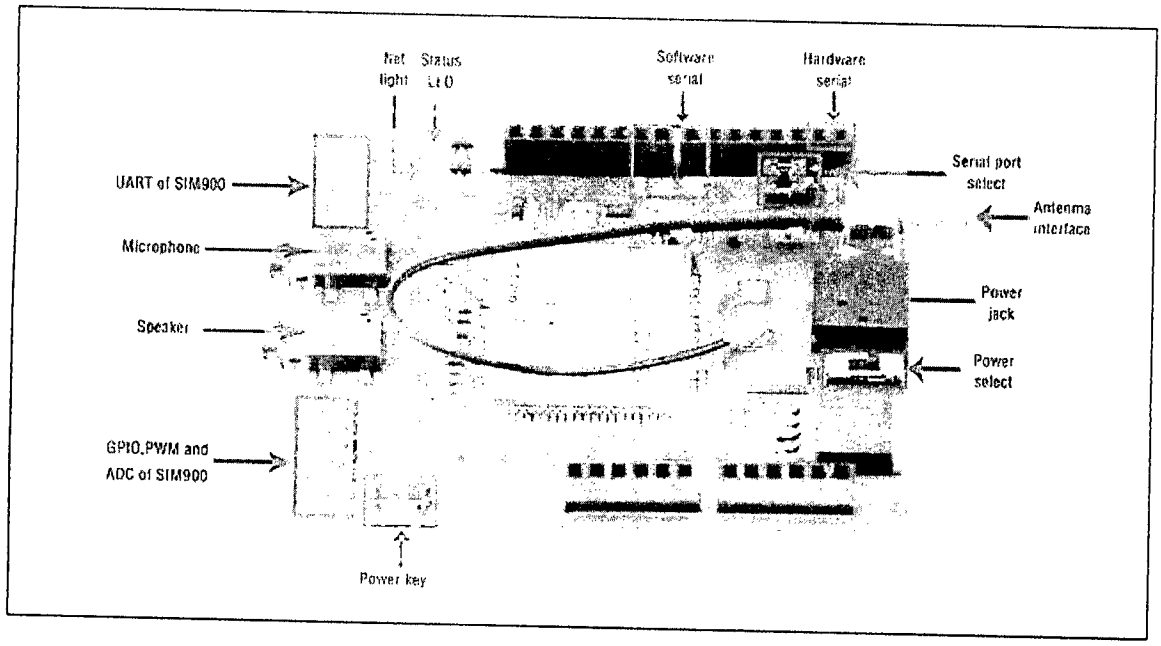


Figura 3.17 Descripción de conexiones del modem SIM900

En la Figura 3.18 se muestra la conexión del Modem SIM900 con el microcontrolador, para lo cual se utilizó el puerto B.6 para transmisión y el puerto B.7 para la recepción.

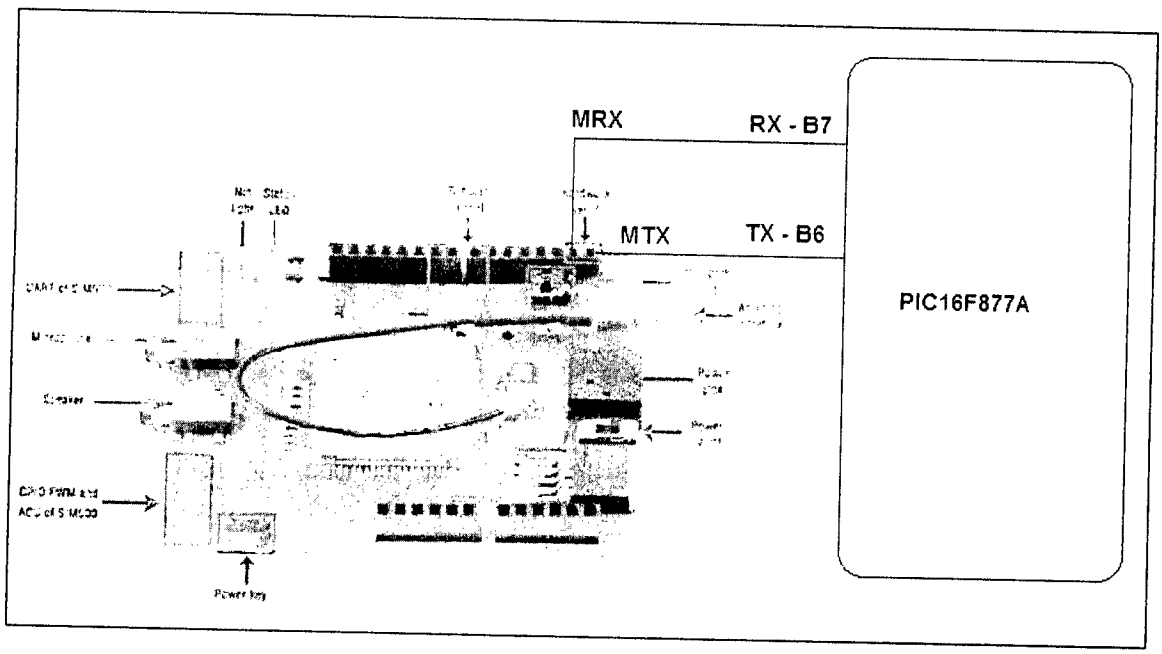


Figura 3.18. Conexiones del Modem GSM con microcontrolador

En la Figura 3.19 se muestras las conexiones del modem con el microcontrolador PIC16F877A, el sistema de perifoneo y la sirena.

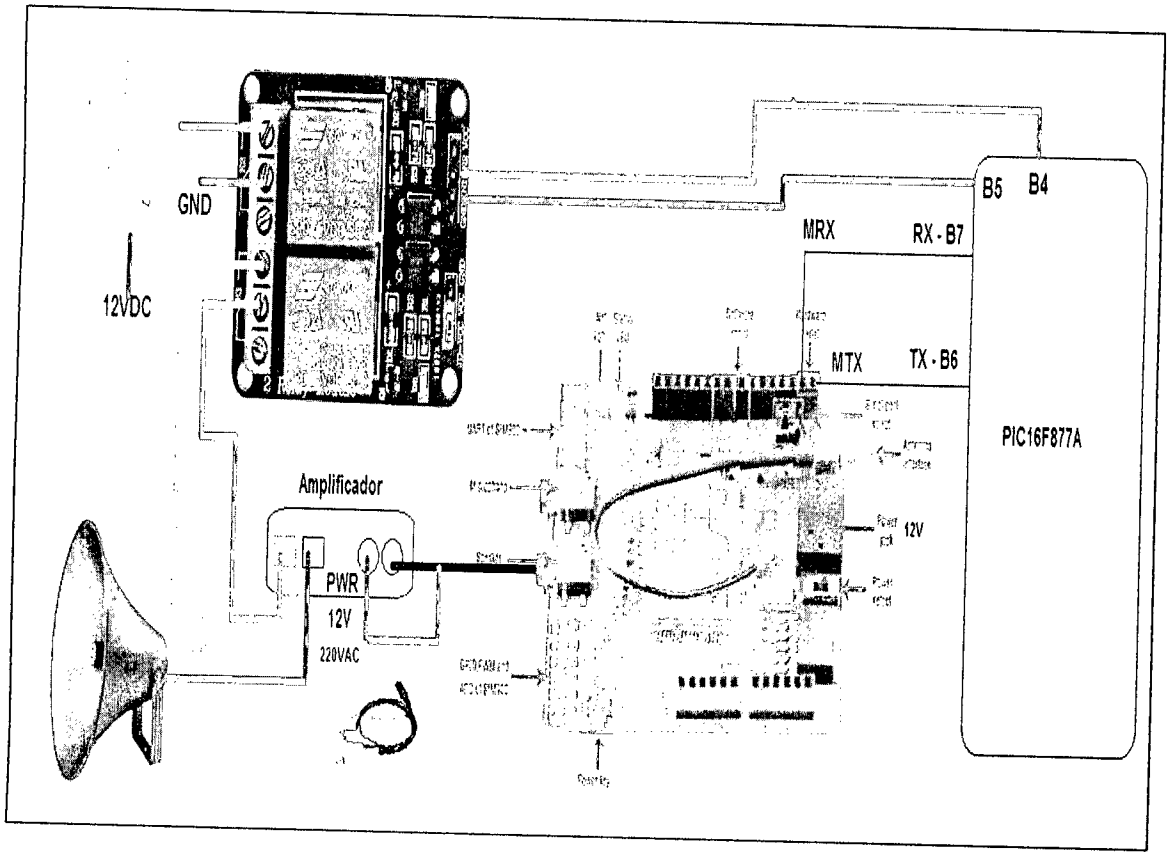


Figura 3.19 Conexiones del modem con el microcontrolador la sirena y bocina

La sirena y la bocina se activa mediante un comando que es el de detector de tonos "DTMF", se utiliza los tonos de los números 1 y 4 para activar sirena, 2 y 5 para bocina.

El comando que se utiliza es el **AT+DDET** en la Figura 3.20 se muestra su formato y descripción.

Para Activar el Detector de tonos se envía el siguiente comando:

AT+DDET=1

Ahora cuando el modem recibe una llamada y se establece la conexión y recibe los tonos entre el (0 1 2 3 4 5 6 7 8 9 * #), el modem envía una cadena de datos como la que se muestra por su puerto serial y es leído por el microcontrolador. Las respuestas por tonos es como sigue:

+DTMF: 0

+DTMF: 1

+DTMF: 2

+DTMF: 3

+DTMF: 4

+DTMF: 5

+DTMF: 6

+DTMF: 7

+DTMF: 8

+DTMF: 9

+DTMF: *

+DTMF: #

AT+DDET

AT+DDET	
Write Command AT+DDET=<status> >	Response OK or ERROR or +CME ERROR: <err> Parameters <status> 0: enable DTMF detection 1: disable DTMF detection
Reference	Note ● This command only can be set when sim card is available and cpin is ready. ● This command cannot be set during a call. If you want use this function, we suggest setting this command before dialing a call. ● Switch of f Sidetone algorithm may benefit to the accuracy of DTMF detection. ● When a call is connected, DTMF detection will start in 300ms ● "1,2,3,4,5,6,7,8,9,0,#,*" only these DTMF strings can be detected

Figura 3.20 Comando AT+DDET para DTMF

En la Figura 3.21 se muestra las líneas de código para configurar el modem SIM900 al encender la tarjeta electrónica de la alarma comunitaria.

En el apéndice se encuentra listado del programa del microcontrolador.

```
PROG_MODEM:
    SEROUT2 TX,84,["AT+IPR=9600",13] 'VELOCIDD DE 9600
    PAUSE 2000
    SEROUT2 TX,84,["AT+IFC=0,0",13] 'RECIBE DIRECTAMENTE A PUERTO SERIAL LOS SMS
    SERIN2 RX,84,2000,PROG_MODEM,[WAIT ("OK")]
    SEROUT2 TX,84,["AT+CNMI=1,2,0,0,0",13] 'RECIBE DIRECTAMENTE A PUERTO SERIAL LOS SMS
    SERIN2 RX,84,2000,PROG_MODEM,[WAIT ("OK")]
    SEROUT2 TX,84,["AT+CMGF=1",13]    ' ACTIVA EN MODO TEXTO TX/RX DE SMS
    SERIN2 RX,84,2000,PROG_MODEM,[WAIT ("OK")]
    SEROUT2 TX,84,["AT+DDET=1",13]    'ACTIVA DETECTOR DE TONOS DTMF
    SERIN2 RX,84,2000,PROG_MODEM,[WAIT ("OK")]
    SEROUT2 TX,84,["ATE0",13]         ' ELIMINA ECO
    SERIN2 RX,84,2000,PROG_MODEM,[WAIT ("OK")]
    SEROUT2 TX,84,["ATS0=2",13]       'RESPUESTA AUTOMATICA DE VOZ DEL MODEM  AL 2 RING
    SERIN2 RX,84,2000,PROG_MODEM,[WAIT ("OK")]
    SEROUT2 TX,84,["AT&W",13]   ' GRABA LA CONFIGURACION
    SERIN2 RX,84,2000,PROG_MODEM,[WAIT ("OK")]
    LCDOUT $fe, 1
    LCDOUT "MODEM PROGRAMADO"
    PAUSE 1000
```

Figura 3.21 Configuración del Modem SIM900

3.2 DESARROLLO DE LA APLICATIVO MOVIL EN SISTEMA OPERATIVO ANDROID

Para el proyecto se ha desarrollado una aplicación para plataforma ANDROID, ha sido desarrollado utilizando el APPINVENTOR 2.

En Este capítulo se mostrara las pantallas principales de la aplicación y se describirá sus principales funciones el código se encuentra en el CD que se adjunta con la Tesis.

En la Figura 3.22 Se muestra el icono de la aplicación SCS (Sistema de Seguridad Ciudadana).

En la Figura 3.23 se muestra la pantalla de aplicaciones de un celular donde se visualizar el icono de la aplicación SCS.

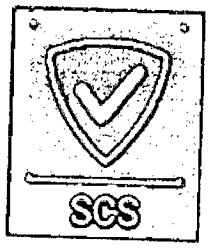


Figura 3.22 Icono de la aplicación

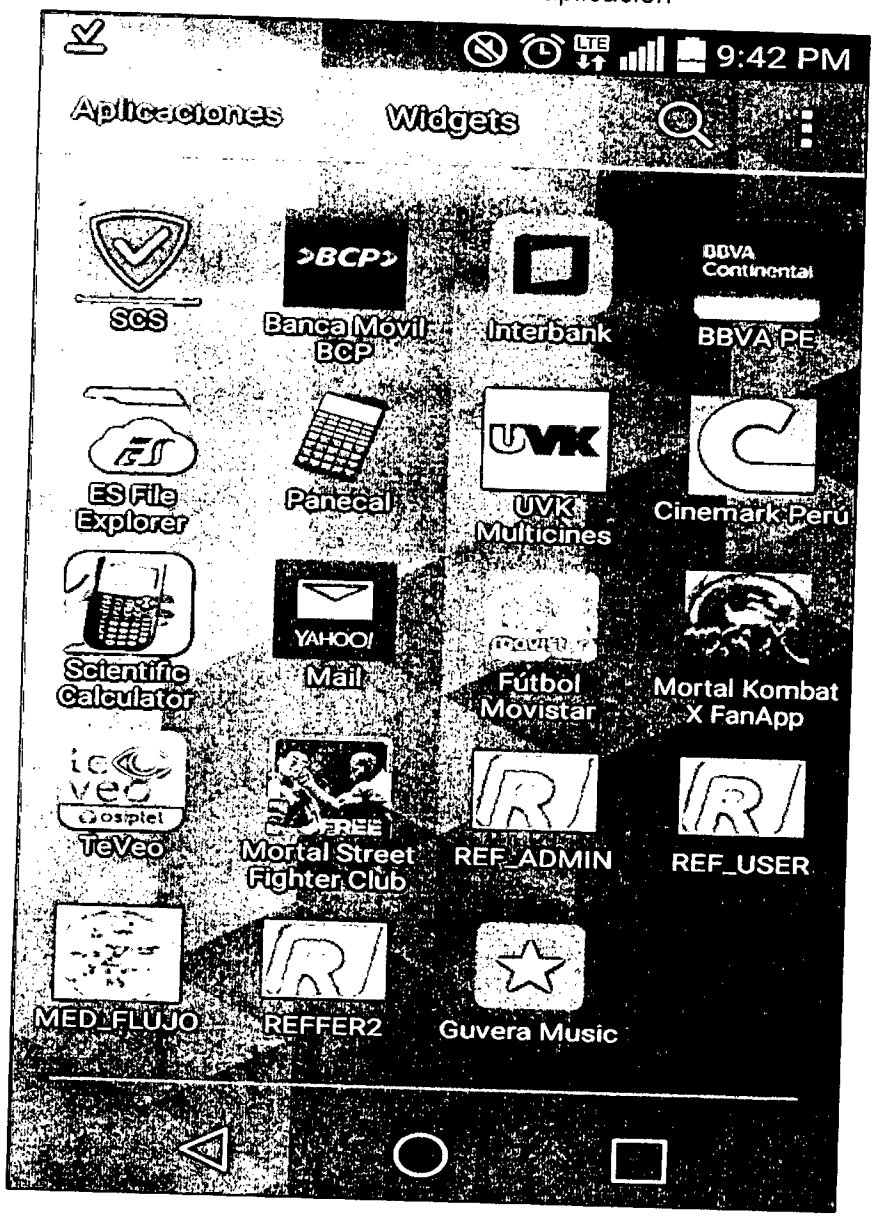


Figura 3.23 Pantalla de aplicaciones de un celular

En la Figura 3.24 se muestra la ´pantalla de configuraciones de los diferentes números de celulares o fijos que usaría la aplicación.

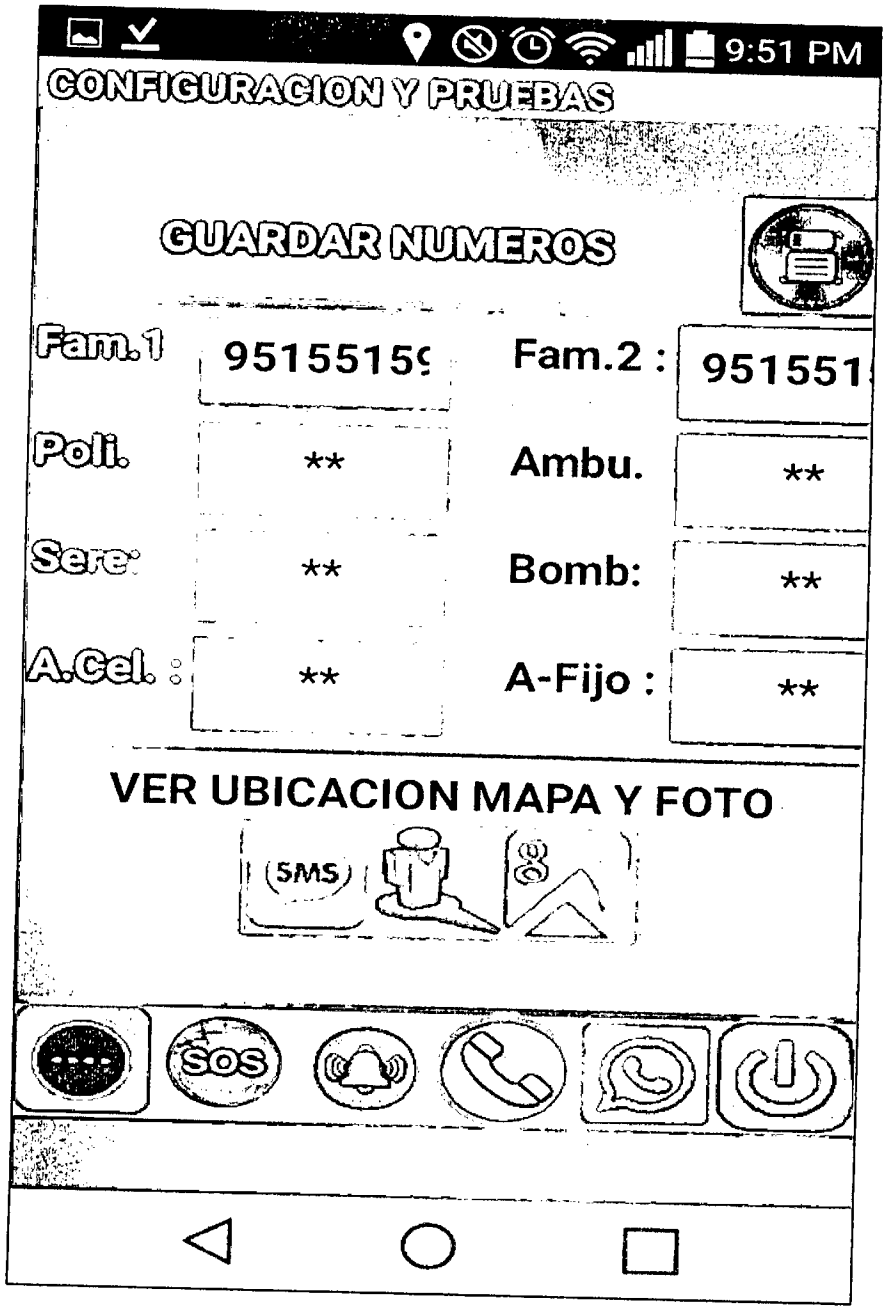


Figura 3.24 Pantalla de configuración de números de celulares de la aplicación

En la Figura 3.25 se muestra la pantalla Emergencia/Pánico en donde se muestra el Botón de pánico que al presionar envía un mensaje de texto a dos familiares y/0 policía, este mensaje contiene las coordenadas de la Ubicación y enlaces con la ubicación y foto referencial de los mapas del google, Ver Figura 3.26

Figura 3.25 Botón de pánico

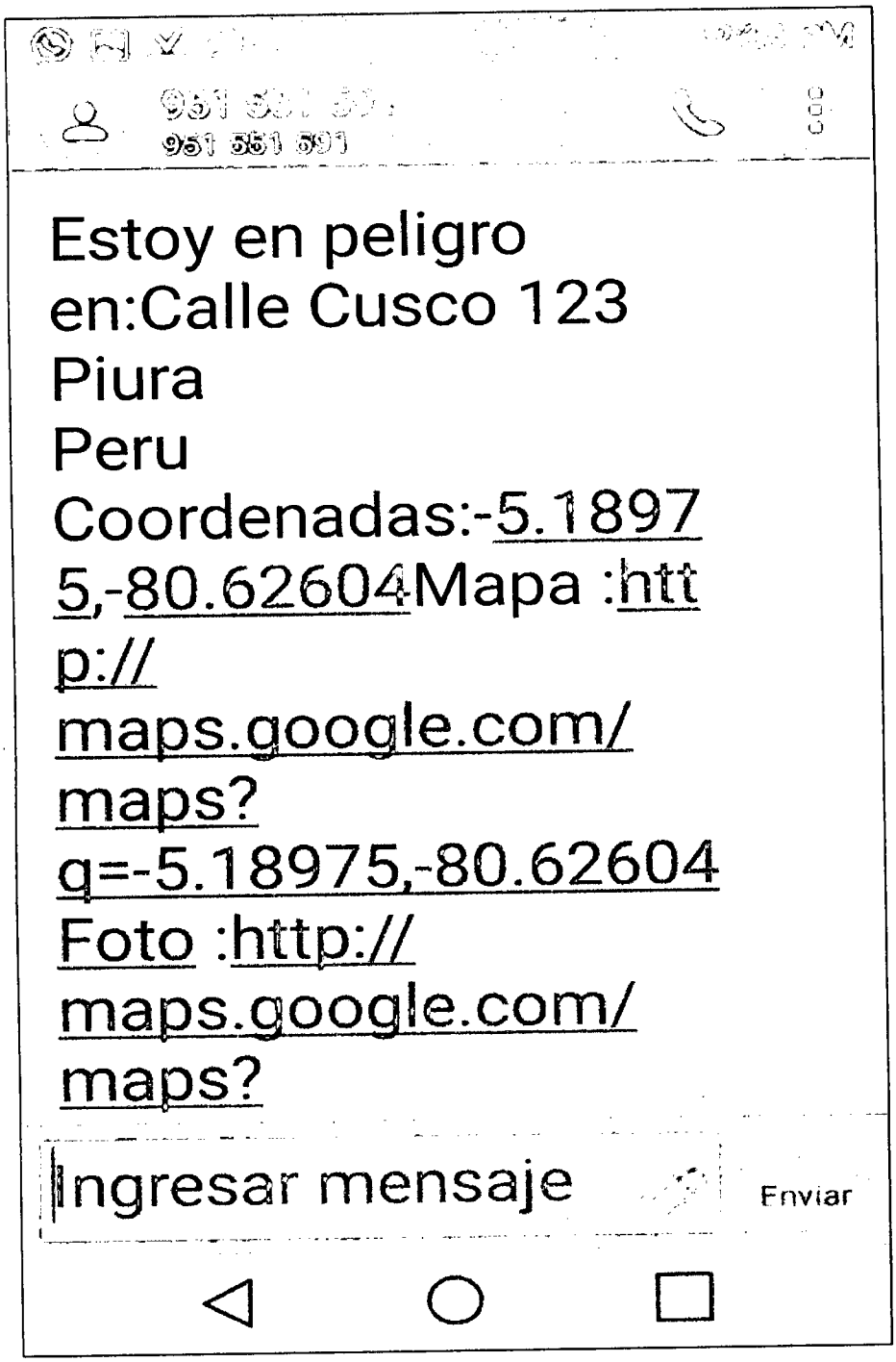


Figura 3.26 Mensaje recibido al presionar el botón de panico

La aplicación también cuenta con una pantalla de directorio de emergencia, el cual permite, realizar llamadas de emergencia a la policía, serenazgo, bomberos y ambulancia Ver Figura 3.26

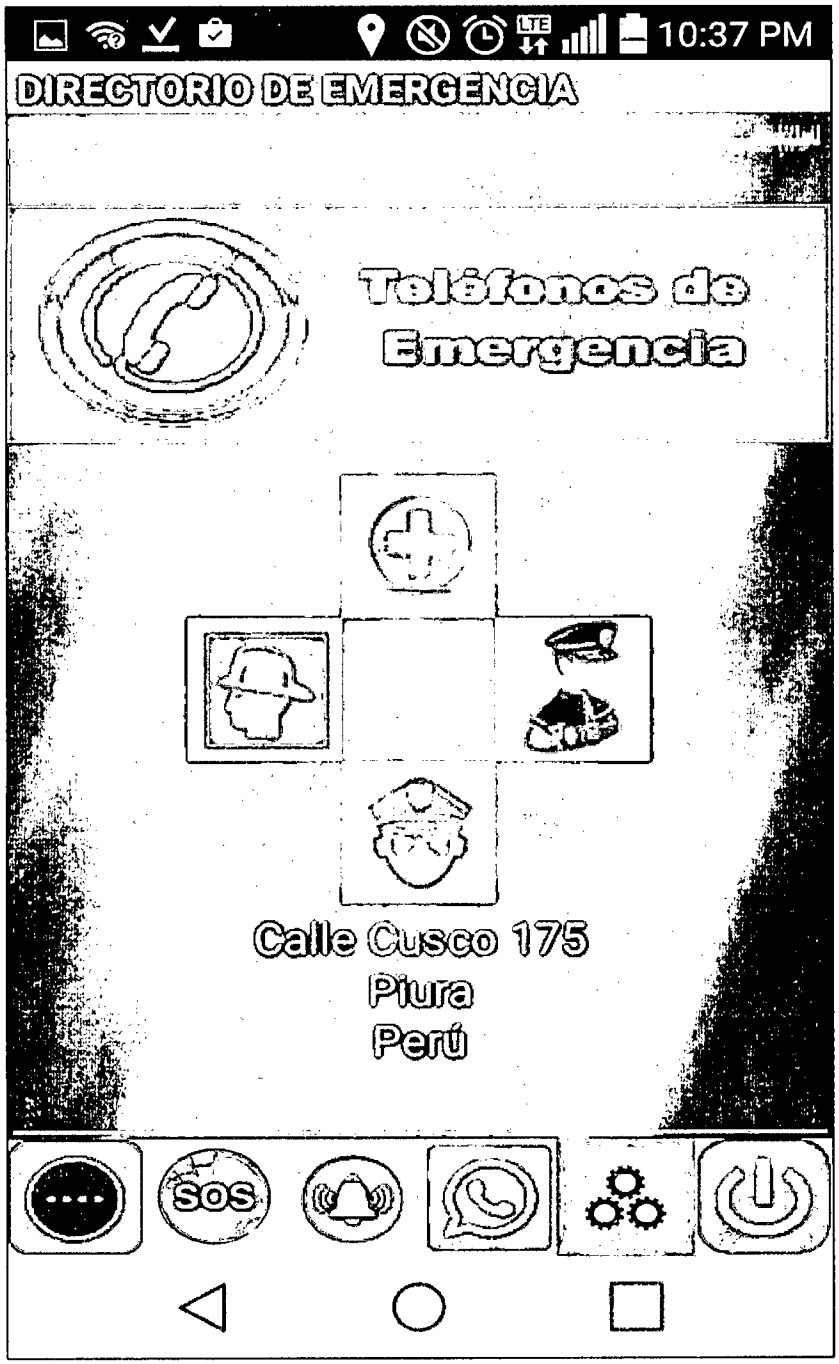


Figura 3.26 Directorio de Emergencia

La aplicación también cuenta con una pantalla desde donde se puede hacer denuncias o avisos a una central sobre diferente tipos de delitos, descuidos o emergencias.

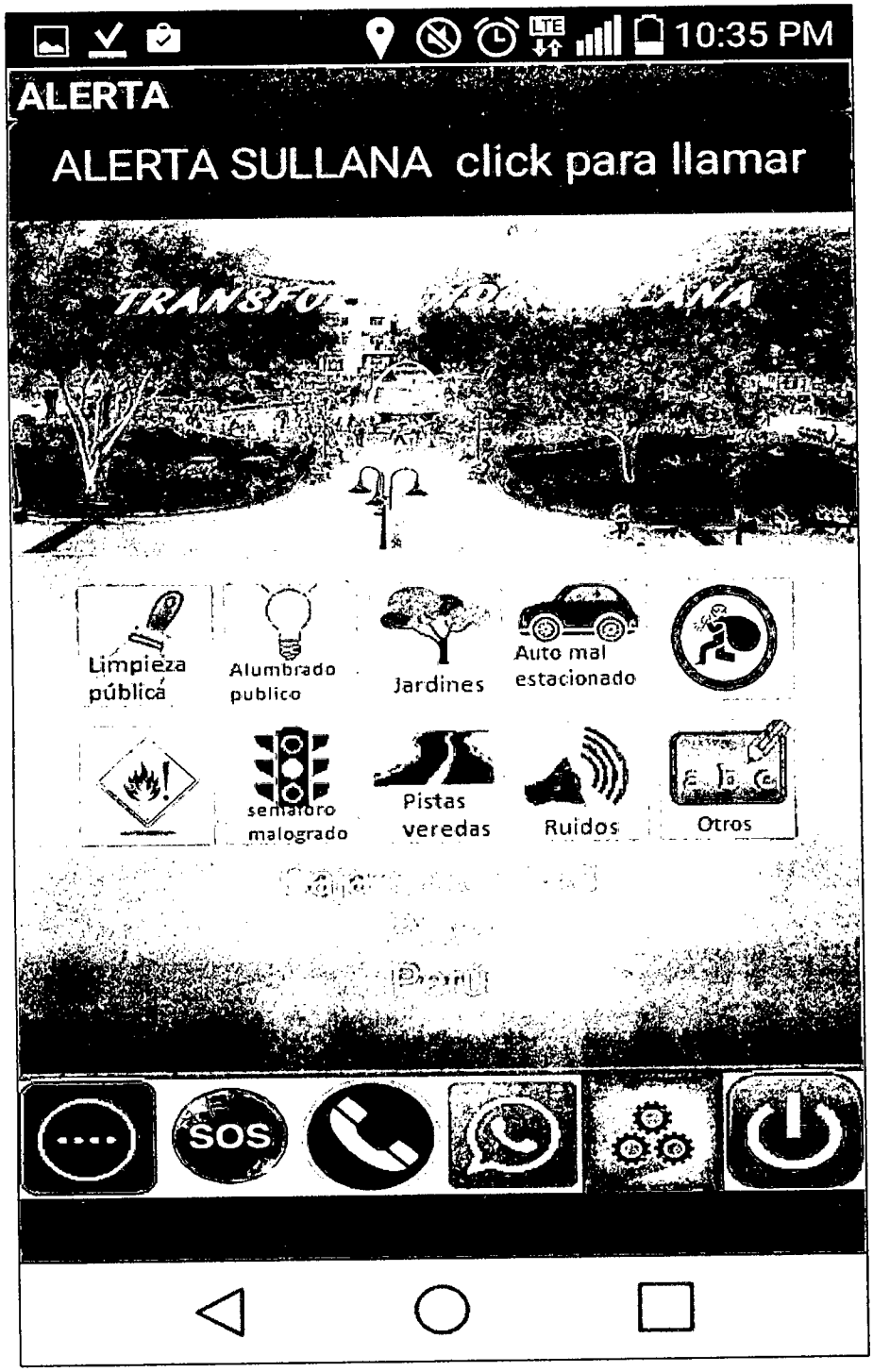


Figura 3.27 Pantalla para denuncias

También se cuenta con la pantalla que permite enviar un mensaje y ubicación por WhatsApp Ver Figura 3.28. En la Figura 3.29 se muestra el mensaje recibido por WhatsApp y al hacer click en los enlaces se mostrara la ubicación y foto referencial de los mapas de google.

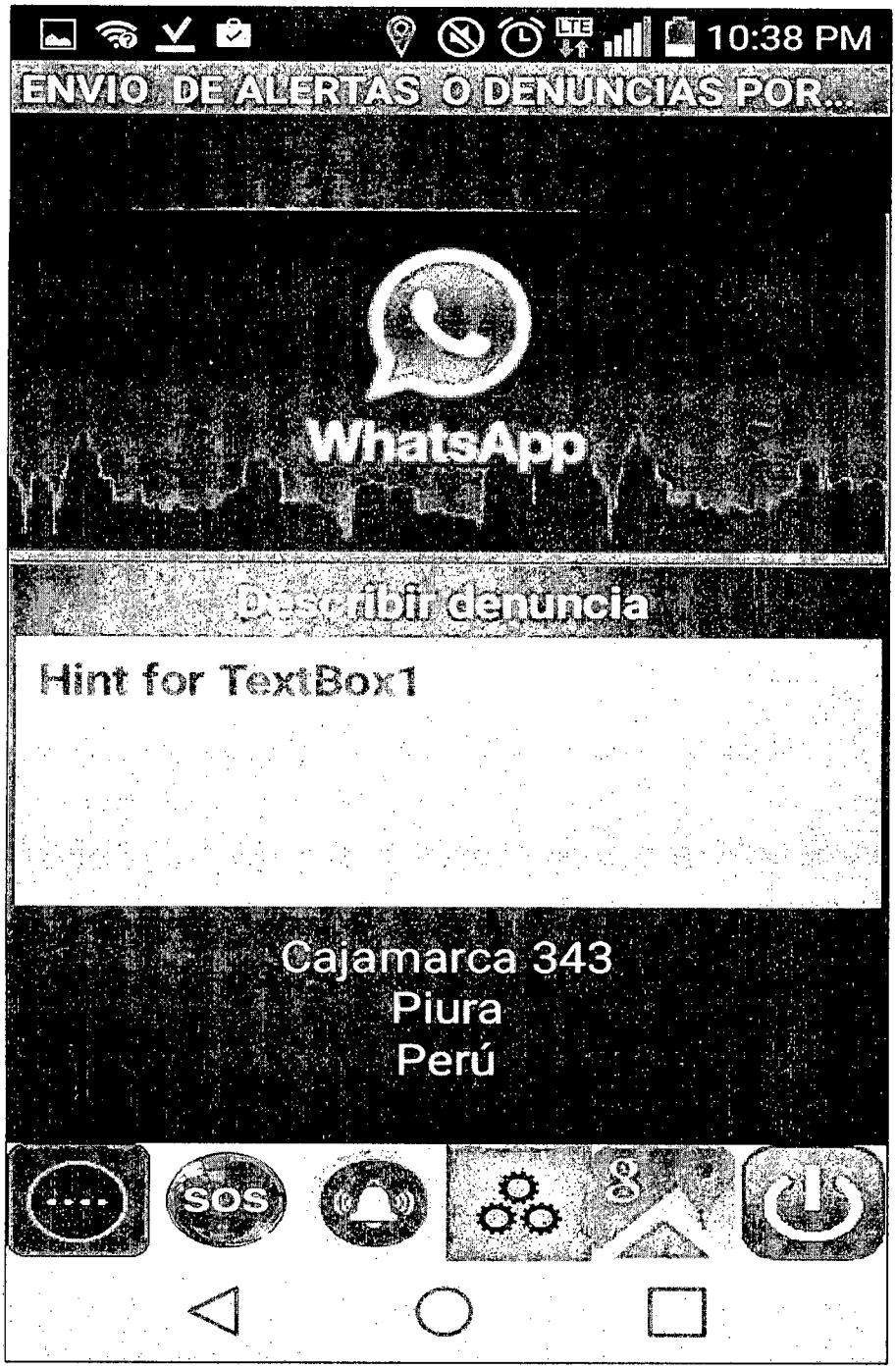


Figura 3.28 Enviar mensajes y ubicación por WhatsAppp

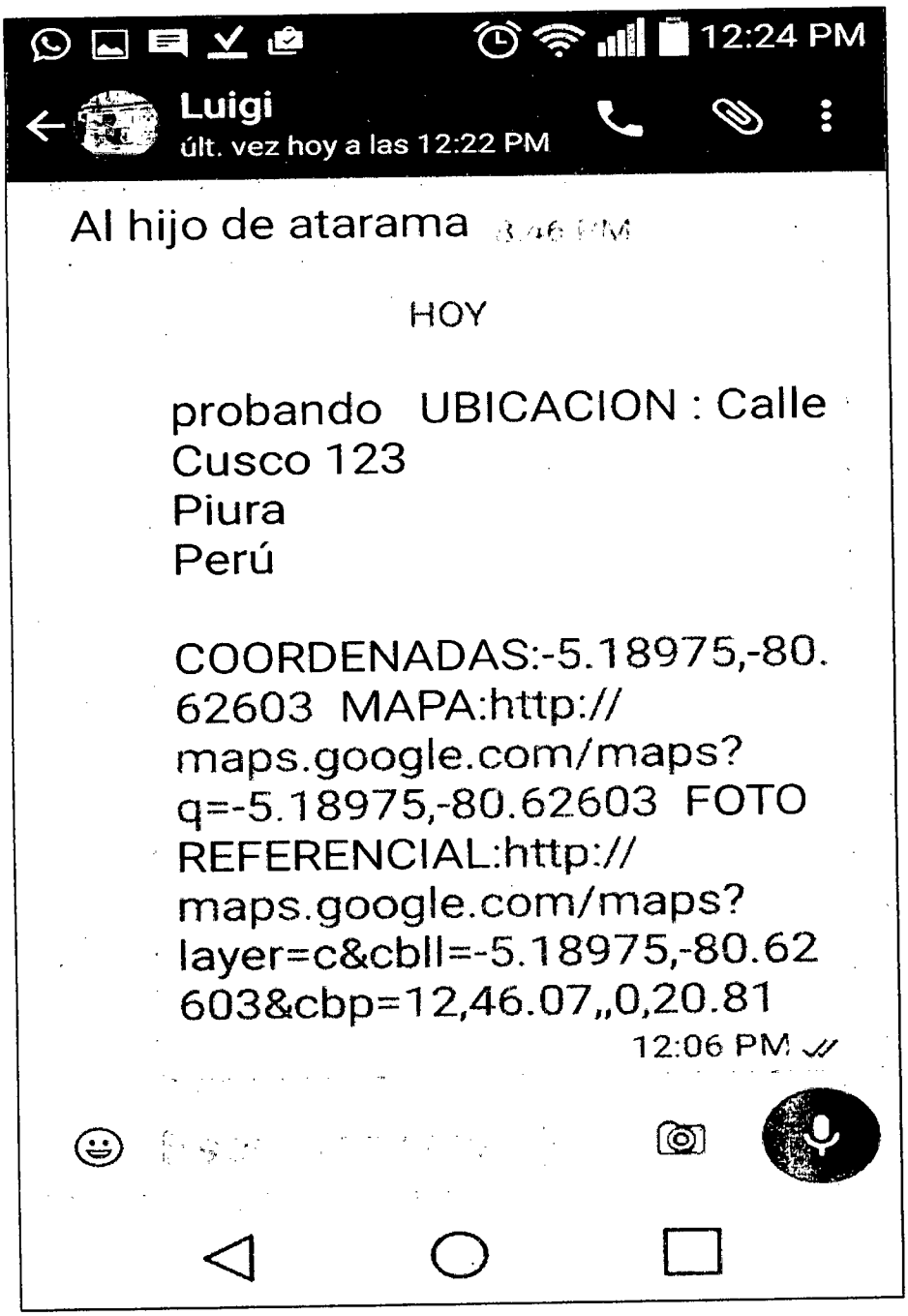


Figura 3.29 Mensaje recibido por whatsApp

Al presionar el enlace de mapa en se visualizara el mapa del googlemaps con la ubicación según coordenadas. Ver Figura 3.30.

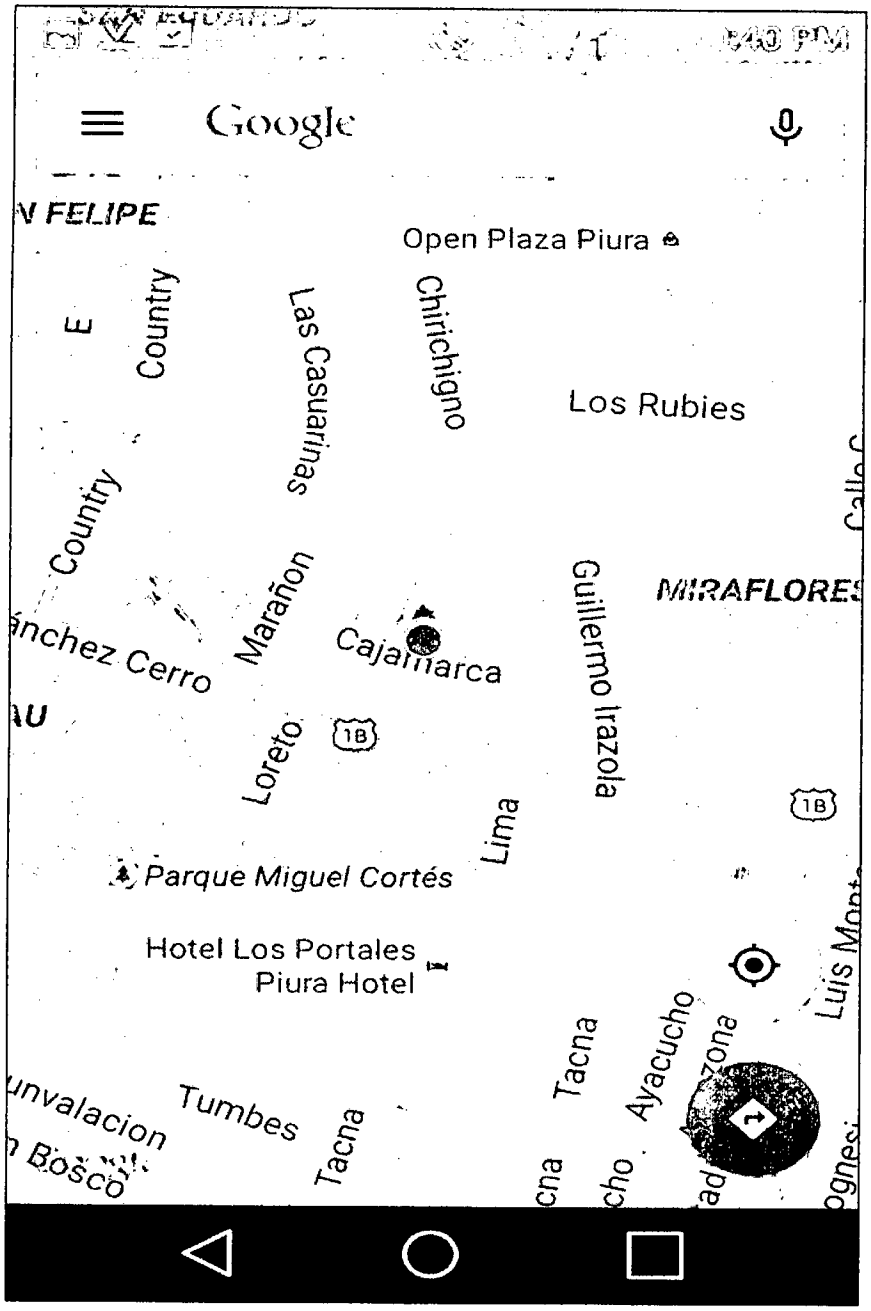


Figura 3.30 ubicación en mapa de googlemaps

Al presionar el enlace de foto en se visualizara la foto referencial del googlemaps según coordenadas. Ver Figura 3.31.

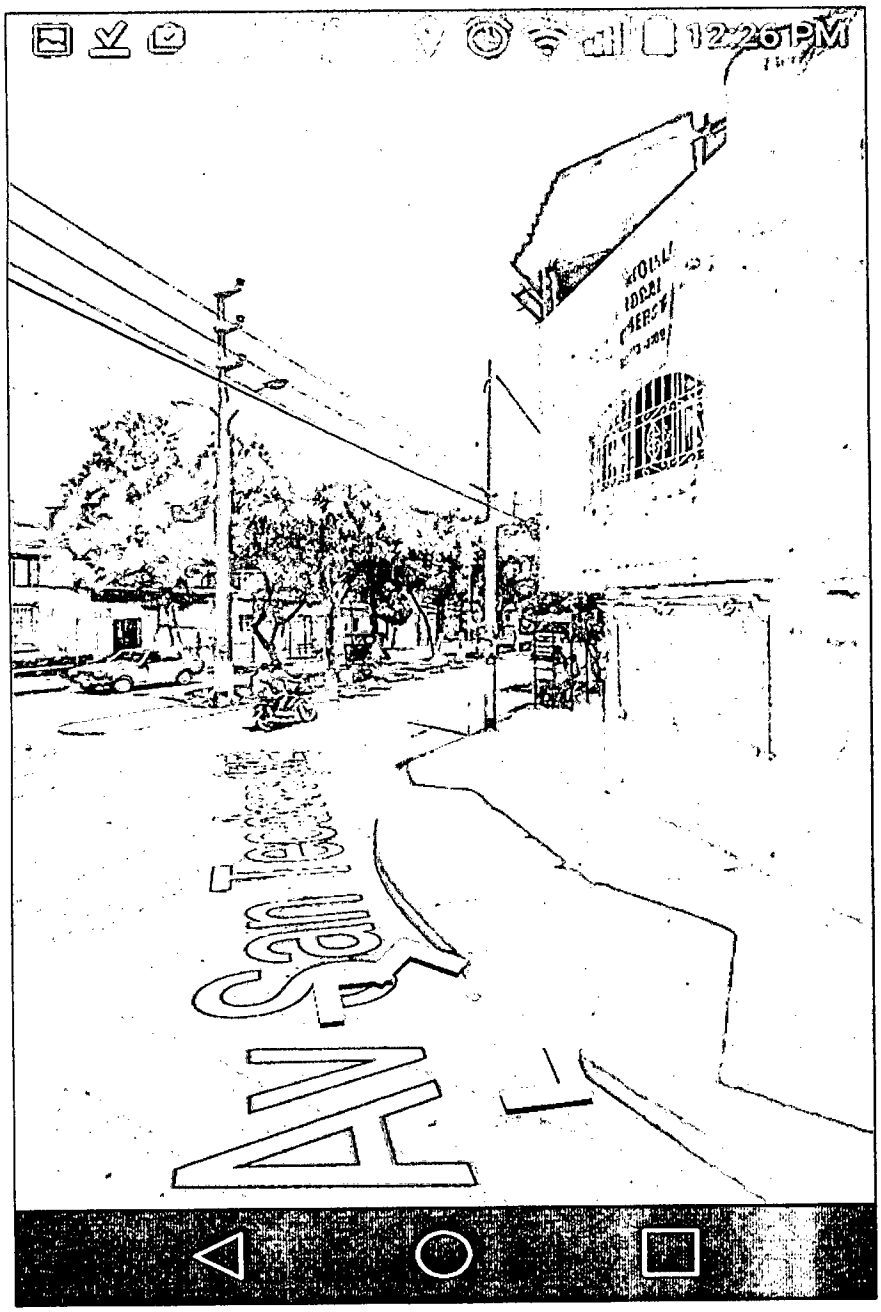


Figura 3.31 Foto referencial de la Ubicación

CAPITULO IV

4.1 COSTOS DEL PROYECTO

En este capítulo se mostrara la tabla de costos para la implementación del proyecto en una se mostrara solo los costos de materiales y la otra tabla se considera los costos de ingeniería con un costo por hora de S/.50.

Tabla 4.1 Costo de materiales

ITEM	MATERIALES	CANTIDAD	P.UNITARIO	P. TOTAL
1	MICROONTROLADOR PIC16F877A	1	S/. 20.00	S/. 20.00
2	LCD 16 X2	1	S/. 22.00	S/. 22.00
3	FUENTE 12 V 5A 60W	1	S/. 65.00	S/. 65.00
4	MODULO 2 RELES	1	S/. 15.00	S/. 15.00
5	MODULO GSM SIM900	1	S/. 170.00	S/. 170.00
6	RESISTENCIAS	10	S/. 0.30	S/. 3.00
7	LEDS	5	S/. 0.40	S/. 2.00
8	CRISTAL	1	S/. 2.50	S/. 2.50
9	SIRENA	2	S/. 60.00	S/. 120.00
10	BOCINA	1	S/. 80.00	S/. 80.00
11	AMPLIFICADOR	1	S/. 135.00	S/. 135.00
12	PLACA IMPRESA	1	S/. 80.00	S/. 80.00
13	CAJA SOLERA PARA PLACA	1	S/. 40.00	S/. 40.00
14	CONECTORES/CABLES	1	S/. 30.00	S/. 30.00
	TOTAL			**S/. 784.50**

Tabla 4.2. Costos de materiales más Ingeniería

ITEM	MATERIALES	CANTIDAD	P.UNITARIO	P. TOTAL
1	MICROONTROLADOR PIC16F877A	1	S/. 20.00	S/. 20.00
2	LCD 16 X2	1	S/. 22.00	S/. 22.00
3	FUENTE 12 V 5A 60W	1	S/. 65.00	S/. 65.00
4	MODULO 2 RELES	1	S/. 15.00	S/. 15.00
5	MODULO GSM SIM900	1	S/. 170.00	S/. 170.00
6	RESISTENCIAS	10	S/. 0.30	S/. 3.00
7	LEDS	5	S/. 0.40	S/. 2.00
8	CRISTAL	1	S/. 2.50	S/. 2.50
9	SIRENA	2	S/. 60.00	S/. 120.00
10	BOCINA	1	S/. 80.00	S/. 80.00
11	AMPLIFICADOR	1	S/. 135.00	S/. 135.00
12	PLACA IMPRESA	1	S/. 80.00	S/. 80.00
13	CAJA SOLERA PARA PLACA	1	S/. 40.00	S/. 40.00
14	CONECTORES/CABLES	1	S/. 30.00	S/. 30.00
19	INGENIERIA Y DESARROLLO DE SOFTWARE COSTO POR HORA	100	S/. 50.00	S/. 5,000.00
	TOTAL			**S/. 5,784.50**

CONCLUSIONES

1. Se logró diseñar un sistema de seguridad ciudadana conformado por una alarma Comunitaria más un aplicativo Android para su funcionamiento complementario usando las Tecnologías de la Información para la prevención de delitos contra las personas y bienes y poder contribuir con la seguridad en la ciudad
2. El reto de los municipios, como gobiernos locales, es contribuir a la solución de los problemas derivados de la inseguridad y la violencia, a través de políticas que supongan "voluntad política, actitudes favorables al proceso, liderazgo institucional, desarrollo de capacidades, cambios organizacionales, inversión social en seguridad ciudadana entre las principales condiciones".[14]
3. La sociedad civil, por su parte, a través de la organización e involucramiento en las actividades, contribuye a solucionar el problema. La participación es un aspecto clave, sin que esto signifique que la ciudadanía sea la responsable de mantener la seguridad, sino más bien, que forme parte activa de las acciones que se emprendan y se sienta corresponsable, junto con la demás instancias involucradas, del mejoramiento de la seguridad y convivencia.
4. Para que el Sistema de Alarmas Comunitarias funcione adecuadamente es fundamental que los barrios estén bien organizados y bajo un liderazgo reconocido. Si bien la tecnología es un apoyo importante, la responsabilidad mayor recae en la capacidad organizativa de éstos.
5. Resulta necesario integrar los centros de ayuda inmediata como la Policía Nacional, Cuerpo de Bomberos, paramédicos, entre otros, para tener una respuesta conjunta.
6. La capacitación sobre el buen uso de las alarmas comunitarias resulta imprescindible, ya que se entenderá que la confiabilidad de las alarmas depende de la forma de uso por parte de los moradores del barrio.
7. Las TIC alternativas juegan un rol clave para facilitar el acceso a información de forma oportuna e inmediata y, a su vez, permiten tomar acciones relevantes en función de la información transferida.

RECOMENDACIONES

- El diseño y construcción, hardware y software deben ir de la mano; de preferencia se recomienda realizar el diseño completo del hardware para luego tener una idea más clara de cómo desarrollar un programa (software) óptimo. Así mismo Se recomienda realizar varias pruebas en conjunto para un mejor entendimiento del sistema y un buen desempeño optimo.
- Contar con los diferentes programas, software electrónico y desarrolladores de circuitos completos y actualizados ya que las versiones de prueba retrasan el desarrollo de pruebas del sistema.
- Previamente las entidades como Municipio, Policía y Bomberos deben organizar, Coordinar, estandarizar y difundir los procedimientos de atención de alarmas, inspecciones de barrios y urbanizaciones, apoyo en mantenimiento de equipos, recepción de comentarios, inquietudes o sugerencias, como también, el reglamento para el buen uso del sistema.
- La capacitación en el uso de alarmas comunitarias debe ser constante e ir acorde a las fases:
    1. Antes de la instalación de la alarma, para identificar requisitos y darla a conocer a todos los vecinos del barrio.
    2. Cuando las alarmas se hayan instalado, para que conozcan sobre su uso y las acciones a tomar.
    3. Después de la instalación, para hacer un seguimiento del uso del sistema a las personas que recibieron la capacitación y a aquellas que quieran integrarse.
- Es importante formar responsables zonales de cada barrio, para el cuidado y verificación del funcionamiento de los equipos de alarmas comunitarias, ya que al encontrarse en áreas públicas estos pueden dañarse. La inspección cercana por parte de los propios moradores es una gran ayuda para la administración de los mismos.
- Es necesario formalizar un procedimiento o protocolo en el que se indique las responsabilidades de cada una de las instancias y procesos que se deben seguir.
- El diseño y la implementación de las alarmas (sirenas) instaladas en cada zona deben ser dimensionadas correctamente para evitar problemas a futuro.
- El Sistema de Alarmas Comunitarias surge al identificar una necesidad manifestada por la colectividad y de la búsqueda de soluciones prácticas y

económicas. El desafío es tener una visión amplia de esa necesidad, a fin de brindar una respuesta integral, a través de coordinación interinstitucional, involucramiento ciudadano y acciones complementarias a otras problemáticas.

- Uno de los mayores retos es aprovechar el auge de las Tecnologías de la Información y Comunicación TIC y adaptarlas a la realidad local para resolver problemas sociales.
- Otro desafío para el Municipio es propiciar investigación para mejorar la herramienta de forma permanente, y confiar en la capacidad de profesionales e instancias locales, como universidades, para encontrar soluciones enmarcadas en el contexto.
- En el mercado local (Tiendas electrónicas) algunos componentes no se hayan, por esta razón se debe realizar búsquedas y compras de estos componentes en la red (Tiendas online) con tiempo, ya que se debe considerar el tiempo de demora en el envío y a ello se suma el retraso del proyecto.
- Investigar las diferentes tendencias tecnológicas de actualidad, aplicables al campo de la electrónica, para mejorar el desarrollo e implementación de proyectos y facilitar la comprensión de los diferentes procesos que implica desarrollar un sistema.

BIBLIOGRAFIA

[1] Torres, M. & Paz, Karim. (2010). Métodos de recolección de datos para una Investigación. Guatemala.

[2] Camarero, J. & Rodríguez, P. (2009). Metodología de desarrollo ágil para sistemas móviles-Introducción al desarrollo con Android y el iPhone. España.

[3] Gasca, M. Camargo, L. & Medina, B. (2013). Metodología para el desarrollo de aplicaciones móviles. Colombia.

[4] Trigas, M. (2011). F. Gestión de proyectos informáticos. México.

[5] Peralta, A. (2003). Metodología Scrum. Uruguay.

[6] Erazo, J. (2013). Aplicación para la gestión de proyectos agiles con Scrum. Ecuador.

[7] Querol, J. (2011). Desarrollo de una aplicación distribuida para dispositivos iOS. España.

[8] Ministerio del Interior. (2013). Plan Nacional de Seguridad Ciudadana. Agosto 25,2014, de Ministerio del Interior Sitio web: ttp://www.mininter.gob.pe/pdfs/Plan.Nacional.Seguridad.Ciudadana.2013-2018.pdf

[9] Imaginated. (2010). Metodología Scrum para Móviles. Setiembre 01,2014, de Imaginated Sitio web: http://www.imaginanet.com/scrum-es-una-metodologia- para-la-programacion-de-aplicaciones-moviles-y-web.html#popup:1026

[10] INEI. (2013). Estadísticas de Seguridad ciudadana. Setiembre 02,2014, de INEI Sitio web: http://conasec.mininter.gob.pe/contenidos/userfiles/files/16127.pdf

[11] Alvarez, J. (2009). La delincuencia en Trujillo. Setiembre 05,2014, de Blog Trujillo en la Noticia Sitio web: http://trujilloenlanoticia.blogspot.com/2009/11/la- delincuencia-en-trujillo.html

[12] Instituto Nacional de Estadística e Informática. (2008). índice temático seguridad ciudadana. Setiembre 05,2014, de INEI Sitio web: http://www.inei.gob.pe/estadisticas/indice-tematico/seguridad-ciudadana/

[13]Universidad alas Peruanas. (2011). Ayuda operacionalizacion de variables, dimensiones. Setiembre 10,2014, de UAP Sitio web: http://uap.intechperu.com/Ucarga/AYUDA.OPERACIONALIZACION_VARIABLES.DIM ENSIONES_50757.pdf

[14] Jarrín, O. (2005). Políticas Públicas de Seguridad Ciudadana: Proyecto de Ley de Seguridad y Convivencia Ciudadana (p. 43-46). Quito: FLACSO - Sede Ecuador. Recuperado de: http://www.flacsoandes.org

MICROCHIP

PIC16F87XA

28/40/44-Pin Enhanced Flash Microcontrollers

Devices Included in this Data Sheet:

- PIC16F873A
- PIC16F874A
- PIC16F876A
- PIC16F877A

High-Performance RISC CPU:

- Only 35 single-word instructions to learn
- All single-cycle instructions except for program branches, which are two-cycle
- Operating speed: DC – 20 MHz clock input
  DC – 200 ns instruction cycle
- Up to 8K x 14 words of Flash Program Memory, Up to 368 x 8 bytes of Data Memory (RAM), Up to 256 x 8 bytes of EEPROM Data Memory
- Pinout compatible to other 28-pin or 40/44-pin PIC16CXXX and PIC16FXXX microcontrollers

Peripheral Features:

- Timer0: 8-bit timer/counter with 8-bit prescaler
- Timer1: 16-bit timer/counter with prescaler, can be incremented during Sleep via external crystal/clock
- Timer2: 8-bit timer/counter with 8-bit period register, prescaler and postscaler
- Two Capture, Compare, PWM modules
  - Capture is 16-bit, max. resolution is 12.5 ns
  - Compare is 16-bit, max. resolution is 200 ns
  - PWM max. resolution is 10-bit
- Synchronous Serial Port (SSP) with SPI (Master mode) and I^2C™ (Master/Slave)
- Universal Synchronous Asynchronous Receiver Transmitter (USART/SCI) with 9-bit address detection
- Parallel Slave Port (PSP) – 8 bits wide with external $\overline{RD}$, $\overline{WR}$ and $\overline{CS}$ controls (40/44-pin only)
- Brown-out detection circuitry for Brown-out Reset (BOR)

Analog Features:

- 10-bit, up to 8-channel Analog-to-Digital Converter (A/D)
- Brown-out Reset (BOR)
- Analog Comparator module with:
  - Two analog comparators
  - Programmable on-chip voltage reference (VREF) module
  - Programmable input multiplexing from device inputs and internal voltage reference
  - Comparator outputs are externally accessible

Special Microcontroller Features:

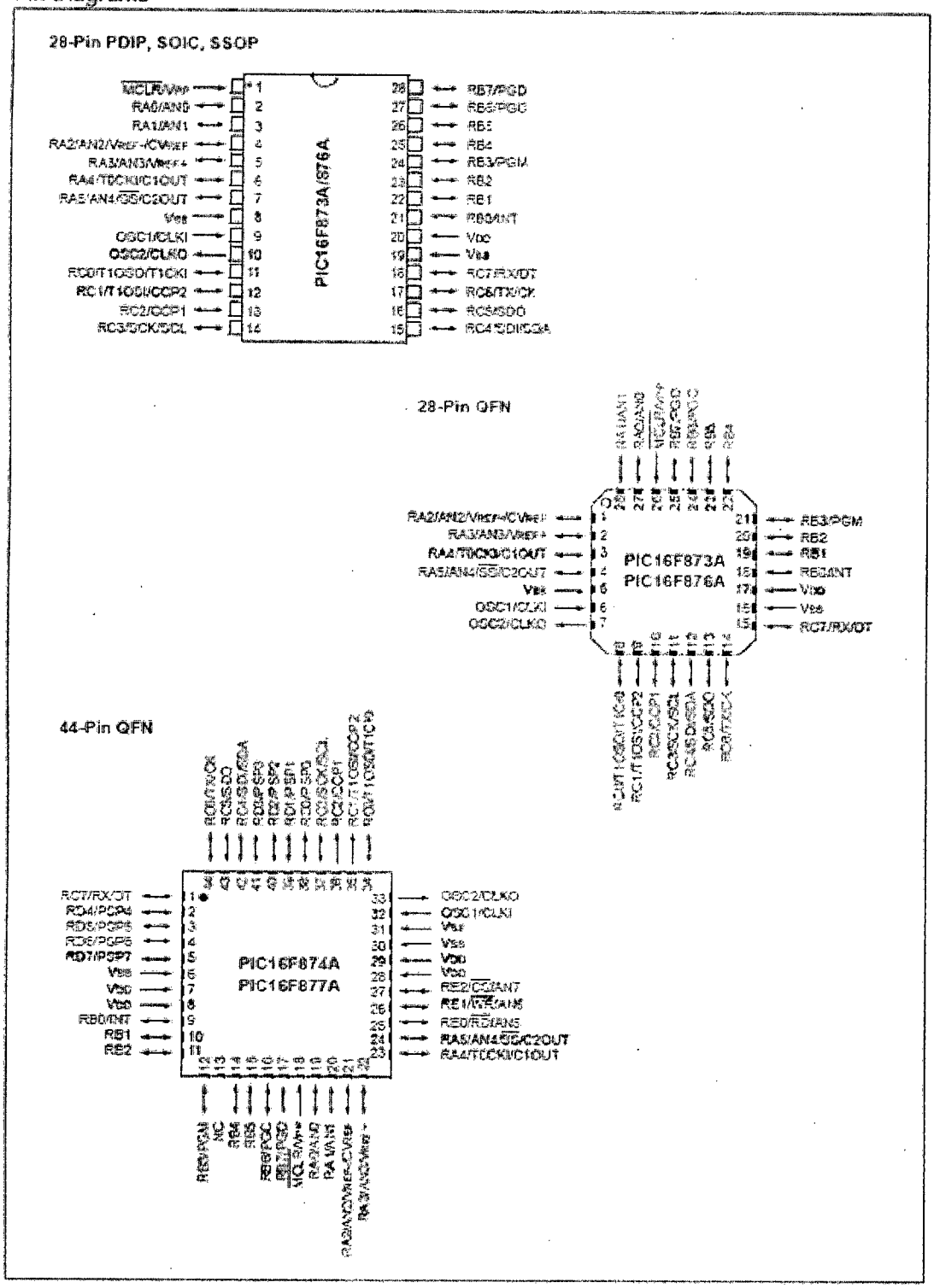
- 100,000 erase/write cycle Enhanced Flash program memory typical
- 1,000,000 erase/write cycle Data EEPROM memory typical
- Data EEPROM Retention > 40 years
- Self-reprogrammable under software control
- In-Circuit Serial Programming™ (ICSP™) via two pins
- Single-supply 5V In-Circuit Serial Programming
- Watchdog Timer (WDT) with its own on-chip RC oscillator for reliable operation
- Programmable code protection
- Power saving Sleep mode
- Selectable oscillator options
- In-Circuit Debug (ICD) via two pins

CMOS Technology:

- Low-power, high-speed Flash/EEPROM technology
- Fully static design
- Wide operating voltage range (2.0V to 5.5V)
- Commercial and Industrial temperature ranges
- Low-power consumption

Device	Program Memory		Data SRAM (Bytes)	EEPROM (Bytes)	I/O	10-bit A/D (ch)	CCP (PWM)	MSSP		USART	Timers 8/16-bit	Comparators
	Bytes	# Single Word Instructions						SPI	Master I²C			
PIC16F873A	7.2K	4096	192	128	22	5	2	Yes	Yes	Yes	2/1	2
PIC16F874A	7.2K	4096	192	128	33	8	2	Yes	Yes	Yes	2/1	2
PIC16F876A	14.3K	8192	368	256	22	5	2	Yes	Yes	Yes	2/1	2
PIC16F877A	14.3K	8192	368	256	33	8	2	Yes	Yes	Yes	2/1	2

Pin Diagrams



Pin Diagrams (Continued)

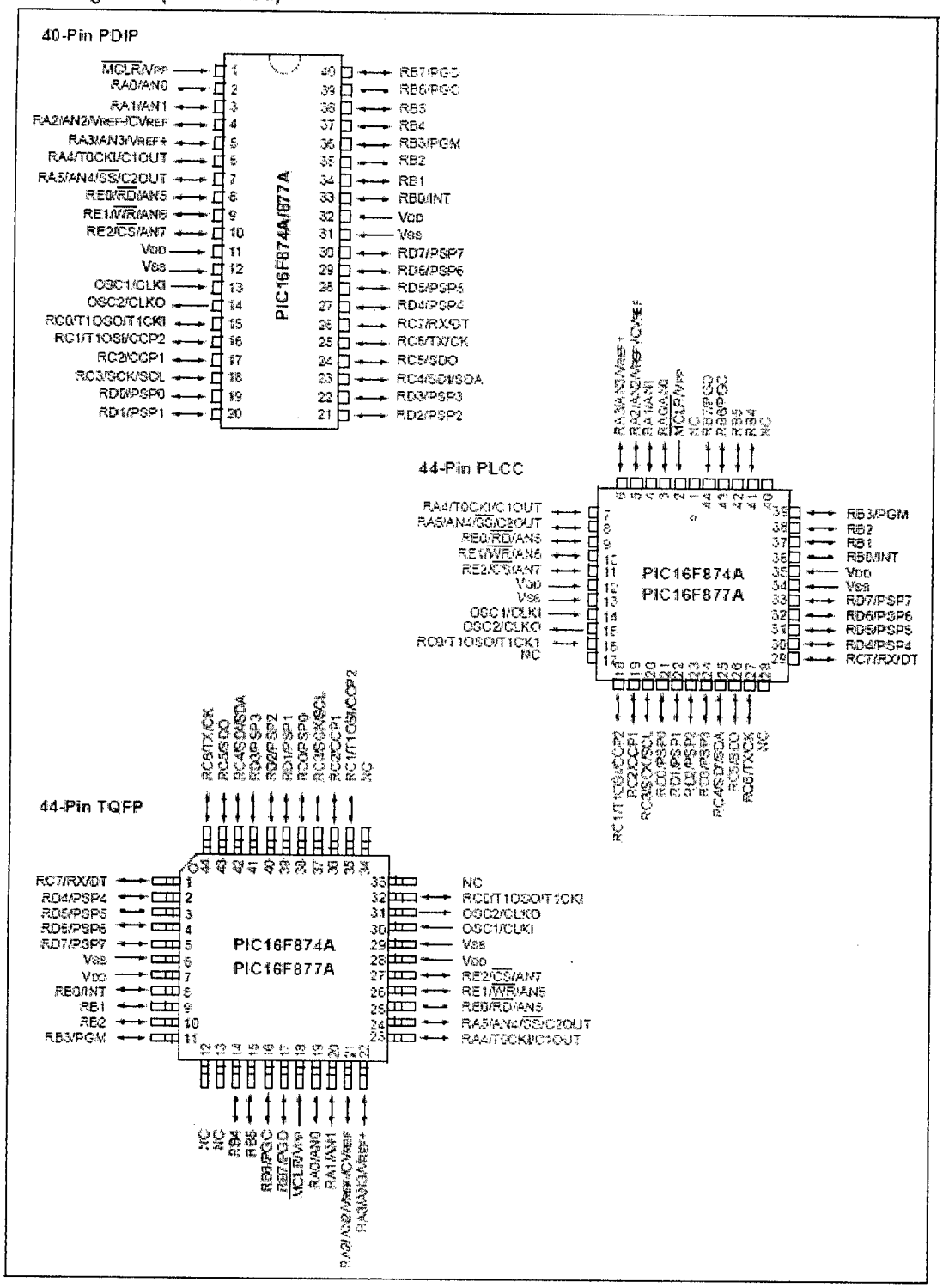
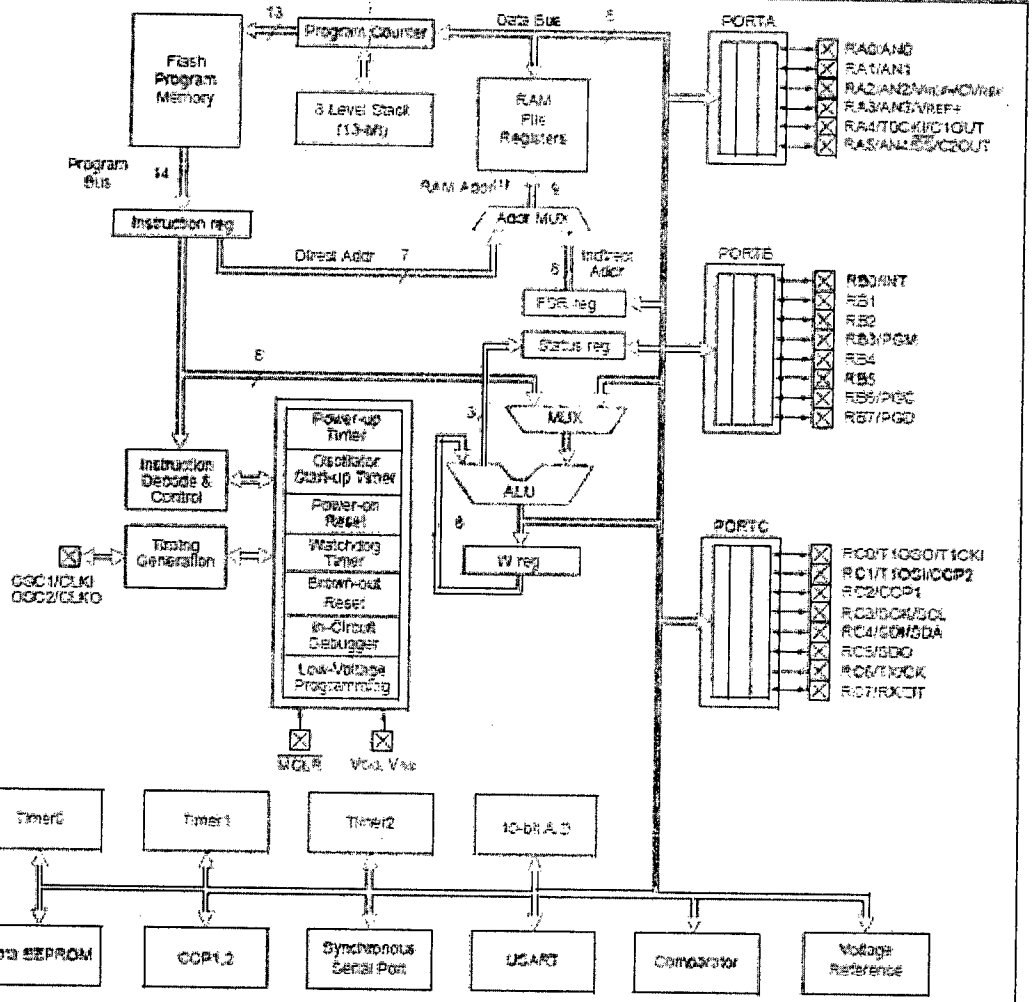


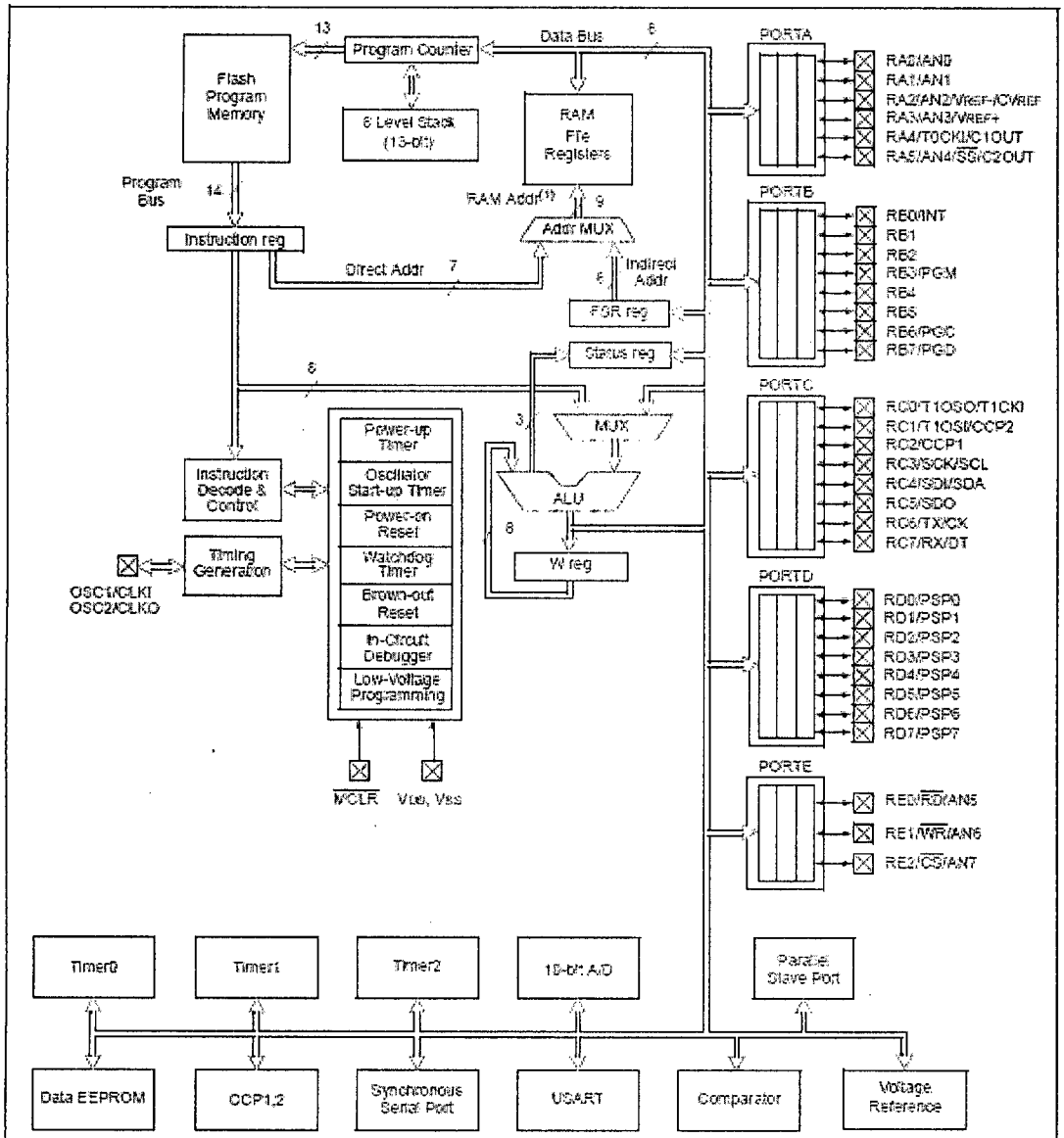
FIGURE 1-1: PIC16F873A/876A BLOCK DIAGRAM



Device	Program Flash	Data Memory	Data EEPROM
PIC16F873A	4K words	192 Bytes	128 Bytes
PIC16F876A	8K words	368 Bytes	256 Bytes

Note 1: Higher order bits are from the Status register.

FIGURE 1-2: PIC16F874A/877A BLOCK DIAGRAM



Device	Program Flash	Data Memory	Data EEPROM
PIC16F874A	4K words	192 Bytes	128 Bytes
PIC16F877A	8K words	368 Bytes	256 Bytes

Note 1: Higher order bits are from the Status register.

TABLE 1-2: PIC16F873A/876A PINOUT DESCRIPTION

Pin Name	PDIP, SOIC, SSOP Pin#	QFN Pin#	I/O/P Type	Buffer Type	Description
OSC1/CLKI	9	6		ST/CMOS(3)	Oscillator crystal or external clock input.
OSC1			I		Oscillator crystal input or external clock source input. ST buffer when configured in RC mode; otherwise CMOS.
CLKI			I		External clock source input. Always associated with pin function OSC1 (see OSC1/CLKI, OSC2/CLKO pins).
OSC2/CLKO	10	7		—	Oscillator crystal or clock output.
OSC2			O		Oscillator crystal output. Connects to crystal or resonator in Crystal Oscillator mode.
CLKO			O		In RC mode, OSC2 pin outputs CLKO, which has 1/4 the frequency of OSC1 and denotes the instruction cycle rate.
MCLR/VPP	1	26		ST	Master Clear (input) or programming voltage (output).
MCLR			I		Master Clear (Reset) input. This pin is an active low Reset to the device.
VPP			P		Programming voltage input.
					PORTA is a bidirectional I/O port.
RA0/AN0	2	27		TTL	
RA0			I/O		Digital I/O.
AN0			I		Analog input 0.
RA1/AN1	3	28		TTL	
RA1			I/O		Digital I/O.
AN1			I		Analog input 1.
RA2/AN2/VREF-/CVREF	4	1		TTL	
RA2			I/O		Digital I/O.
AN2			I		Analog input 2.
VREF-			I		A/D reference voltage (Low) input.
CVREF			O		Comparator VREF output.
RA3/AN3/VREF+	5	2		TTL	
RA3			I/O		Digital I/O.
AN3			I		Analog input 3.
VREF+			I		A/D reference voltage (High) input.
RA4/T0CKI/C1OUT	6	3		ST	
RA4			I/O		Digital I/O – Open-drain when configured as output.
T0CKI			I		Timer0 external clock input.
C1OUT			O		Comparator 1 output.
RA5/AN4/SS/C2OUT	7	4		TTL	
RA5			I/O		Digital I/O.
AN4			I		Analog input 4.
SS			I		SPI slave select input.
C2OUT			O		Comparator 2 output.

Legend: I = input O = output I/O = input/output P = power
— = Not used TTL = TTL input ST = Schmitt Trigger input

Note 1: This buffer is a Schmitt Trigger input when configured as the external interrupt.
2: This buffer is a Schmitt Trigger input when used in Serial Programming mode.
3: This buffer is a Schmitt Trigger input when configured in RC Oscillator mode and a CMOS input otherwise.

TABLE 1-2: PIC16F873A/876A PINOUT DESCRIPTION (CONTINUED)

Pin Name	PDIP, SOIC, SSOP Pin#	QFN Pin#	I/O/P Type	Buffer Type	Description
					PORTB is a bidirectional I/O port. PORTB can be software programmed for internal weak pull-ups on all inputs.
RB0/INT	21	18		TTL/ST(1)	
RB0			I/O		Digital I/O.
INT			I		External interrupt.
RB1	22	19	I/O	TTL	Digital I/O.
RB2	23	20	I/O	TTL	Digital I/O.
RB3/PGM	24	21		TTL	
RB3			I/O		Digital I/O.
PGM			I		Low-voltage (single-supply) ICSP programming enable pin.
RB4	25	22	I/O	TTL	Digital I/O.
RB5	26	23	I/O	TTL	Digital I/O.
RB6/PGC	27	24		TTL/ST(2)	
RB6			I/O		Digital I/O.
PGC			I		In-circuit debugger and ICSP programming clock.
RB7/PGD	28	25		TTL/ST(2)	
RB7			I/O		Digital I/O.
PGD			I/O		In-circuit debugger and ICSP programming data.
					PORTC is a bidirectional I/O port.
RC0/T1OSO/T1CKI	11	8		ST	
RC0			I/O		Digital I/O.
T1OSO			O		Timer1 oscillator output.
T1CKI			I		Timer1 external clock input.
RC1/T1OSI/CCP2	12	9		ST	
RC1			I/O		Digital I/O.
T1OSI			I		Timer1 oscillator input.
CCP2			I/O		Capture2 input, Compare2 output, PWM2 output.
RC2/CCP1	13	10		ST	
RC2			I/O		Digital I/O.
CCP1			I/O		Capture1 input, Compare1 output, PWM1 output.
RC3/SCK/SCL	14	11		ST	
RC3			I/O		Digital I/O.
SCK			I/O		Synchronous serial clock input/output for SPI mode.
SCL			I/O		Synchronous serial clock input/output for I^2C mode.
RC4/SDI/SDA	15	12		ST	
RC4			I/O		Digital I/O.
SDI			I		SPI data in.
SDA			I/O		I^2C data I/O.
RC5/SDO	16	13		ST	
RC5			I/O		Digital I/O.
SDO			O		SPI data out.
RC6/TX/CK	17	14		ST	
RC6			I/O		Digital I/O.
TX			O		USART asynchronous transmit.
CK			I/O		USART synchronous clock.
RC7/RX/DT	18	15		ST	
RC7			I/O		Digital I/O.
RX			I		USART asynchronous receive.
DT			I/O		USART synchronous data.
Vss	8, 19	5, 6	P	—	Ground reference for logic and I/O pins.
Vdd	20	17	P	—	Positive supply for logic and I/O pins.

Legend: I = input O = output I/O = input/output P = power
— = Not used TTL = TTL input ST = Schmitt Trigger input

Note 1: This buffer is a Schmitt Trigger input when configured as the external interrupt.
2: This buffer is a Schmitt Trigger input when used in Serial Programming mode.
3: This buffer is a Schmitt Trigger input when configured in RC Oscillator mode and a CMOS input otherwise.

TABLE 1-3: PIC16F874A/877A PINOUT DESCRIPTION

Pin Name	PDIP Pin#	PLCC Pin#	TQFP Pin#	QFN Pin#	I/O/P Type	Buffer Type	Description
OSC1/CLKI	13	14	30	32		ST/CMOS(4)	Oscillator crystal or external clock input.
OSC1					I		Oscillator crystal input or external clock source input. ST buffer when configured in RC mode; otherwise CMOS.
CLKI					I		External clock source input. Always associated with pin function OSC1 (see OSC1/CLKI, OSC2/CLKO pins).
OSC2/CLKO	14	15	31	33		—	Oscillator crystal or clock output.
OSC2					O		Oscillator crystal output. Connects to crystal or resonator in Crystal Oscillator mode.
CLKO					O		In RC mode, OSC2 pin outputs CLKO, which has 1/4 the frequency of OSC1 and denotes the instruction cycle rate.
$\overline{MCLR}/V_{PP}$	1	2	18	18		ST	Master Clear (input) or programming voltage (output).
$\overline{MCLR}$					I		Master Clear (Reset) input. This pin is an active low Reset to the device.
V_{PP}					P		Programming voltage input.
							PORTA is a bidirectional I/O port.
RA0/AN0	2	3	19	19		TTL	
RA0					I/O		Digital I/O.
AN0					I		Analog input 0.
RA1/AN1	3	4	20	20		TTL	
RA1					I/O		Digital I/O.
AN1					I		Analog input 1.
RA2/AN2/V_{REF-}/CV_{REF}	4	5	21	21		TTL	
RA2					I/O		Digital I/O.
AN2					I		Analog input 2.
V_{REF-}					I		A/D reference voltage (Low) input.
CV_{REF}					O		Comparator V_{REF} output.
RA3/AN3/V_{REF+}	5	6	22	22		TTL	
RA3					I/O		Digital I/O.
AN3					I		Analog input 3.
V_{REF+}					I		A/D reference voltage (High) input.
RA4/T0CKI/C1OUT	6	7	23	23		ST	
RA4					I/O		Digital I/O – Open-drain when configured as output.
T0CKI					I		Timer0 external clock input.
C1OUT					O		Comparator 1 output.
RA5/AN4/$\overline{SS}$/C2OUT	7	8	24	24		TTL	
RA5					I/O		Digital I/O.
AN4					I		Analog input 4.
$\overline{SS}$					I		SPI slave select input.
C2OUT					O		Comparator 2 output.

Legend: I = input O = output I/O = input/output P = power
— = Not used TTL = TTL input ST = Schmitt Trigger input

Note 1: This buffer is a Schmitt Trigger input when configured as the external interrupt.
2: This buffer is a Schmitt Trigger input when used in Serial Programming mode.
3: This buffer is a Schmitt Trigger input when configured in RC Oscillator mode and a CMOS input otherwise.

TABLE 1-3: PIC16F874A/877A PINOUT DESCRIPTION (CONTINUED)

Pin Name	PDIP Pin#	PLCC Pin#	TQFP Pin#	QFN Pin#	I/O/P Type	Buffer Type	Description
							PORTB is a bidirectional I/O port. PORTB can be software programmed for internal weak pull-up on all inputs.
RB0/INT	33	36	8	9		TTL/ST(1)	
RB0					I/O		Digital I/O.
INT					I		External interrupt.
RB1	34	37	9	10	I/O	TTL	Digital I/O.
RB2	35	38	10	11	I/O	TTL	Digital I/O.
RB3/PGM	36	39	11	12		TTL	
RB3					I/O		Digital I/O.
PGM					I		Low-voltage ICSP programming enable pin.
RB4	37	41	14	14	I/O	TTL	Digital I/O.
RB5	38	42	15	15	I/O	TTL	Digital I/O.
RB6/PGC	39	43	16	16		TTL/ST(2)	
RB6					I/O		Digital I/O.
PGC					I		In-circuit debugger and ICSP programming clock.
RB7/PGD	40	44	17	17		TTL/ST(2)	
RB7					I/O		Digital I/O.
PGD					I/O		In-circuit debugger and ICSP programming data.

Legend: I = input O = output I/O = input/output P = power
— = Not used TTL = TTL input ST = Schmitt Trigger input

Note 1: This buffer is a Schmitt Trigger input when configured as the external interrupt.
2: This buffer is a Schmitt Trigger input when used in Serial Programming mode.
3: This buffer is a Schmitt Trigger input when configured in RC Oscillator mode and a CMOS input otherwise.

TABLE 1-3: PIC16F874A/877A PINOUT DESCRIPTION (CONTINUED)

Pin Name	PDIP Pin#	PLCC Pin#	TQFP Pin#	QFN Pin#	I/O/P Type	Buffer Type	Description
							PORTC is a bidirectional I/O port.
RC0/T1OSO/T1CKI	15	16	32	34		ST	
RC0					I/O		Digital I/O.
T1OSO					O		Timer1 oscillator output.
T1CKI					I		Timer1 external clock input.
RC1/T1OSI/CCP2	16	18	35	35		ST	
RC1					I/O		Digital I/O.
T1OSI					I		Timer1 oscillator input.
CCP2					I/O		Capture2 input, Compare2 output, PWM2 output.
RC2/CCP1	17	19	36	36		ST	
RC2					I/O		Digital I/O.
CCP1					I/O		Capture1 input, Compare1 output, PWM1 output.
RC3/SCK/SCL	18	20	37	37		ST	
RC3					I/O		Digital I/O.
SCK					I/O		Synchronous serial clock input/output for SPI mode.
SCL					I/O		Synchronous serial clock input/output for I²C mode.
RC4/SDI/SDA	23	25	42	42		ST	
RC4					I/O		Digital I/O.
SDI					I		SPI data in.
SDA					I/O		I²C data I/O.
RC5/SDO	24	26	43	43		ST	
RC5					I/O		Digital I/O.
SDO					O		SPI data out.
RC6/TX/CK	25	27	44	44		ST	
RC6					I/O		Digital I/O.
TX					O		USART asynchronous transmit.
CK					I/O		USART1 synchronous clock.
RC7/RX/DT	26	29	1	1		ST	
RC7					I/O		Digital I/O.
RX					I		USART asynchronous receive.
DT					I/O		USART synchronous data.

Legend: I = input O = output I/O = input/output P = power
— = Not used TTL = TTL input ST = Schmitt Trigger input

Note 1: This buffer is a Schmitt Trigger input when configured as the external interrupt.
2: This buffer is a Schmitt Trigger input when used in Serial Programming mode.
3: This buffer is a Schmitt Trigger input when configured in RC Oscillator mode and a CMOS input otherwise.

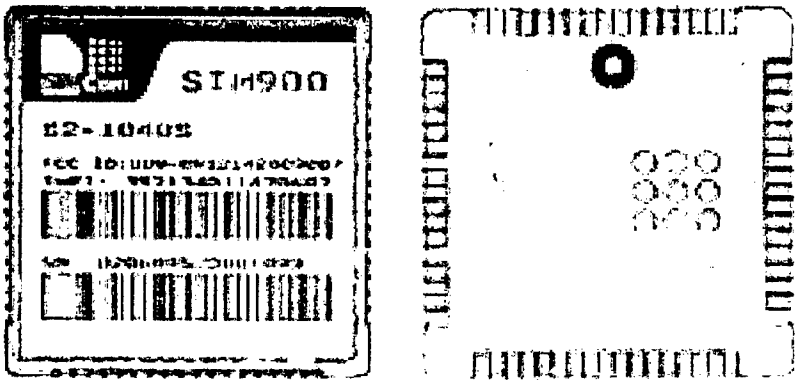
TABLE 1-3: PIC16F874A/877A PINOUT DESCRIPTION (CONTINUED)

Pin Name	PDIP Pin#	PLCC Pin#	TQFP Pin#	QFN Pin#	I/O/P Type	Buffer Type	Description
							PORTD is a bidirectional I/O port or Parallel Slave Port when interfacing to a microprocessor bus.
RD0/PSP0	19	21	38	38		ST/TTL(3)	
RD0					I/O		Digital I/O.
PSP0					I/O		Parallel Slave Port data.
RD1/PSP1	20	22	39	39		ST/TTL(3)	
RD1					I/O		Digital I/O.
PSP1					I/O		Parallel Slave Port data.
RD2/PSP2	21	23	40	40		ST/TTL(3)	
RD2					I/O		Digital I/O.
PSP2					I/O		Parallel Slave Port data.
RD3/PSP3	22	24	41	41		ST/TTL(3)	
RD3					I/O		Digital I/O.
PSP3					I/O		Parallel Slave Port data.
RD4/PSP4	27	30	2	2		ST/TTL(3)	
RD4					I/O		Digital I/O.
PSP4					I/O		Parallel Slave Port data.
RD5/PSP5	28	31	3	3		ST/TTL(3)	
RD5					I/O		Digital I/O.
PSP5					I/O		Parallel Slave Port data.
RD6/PSP6	29	32	4	4		ST/TTL(3)	
RD6					I/O		Digital I/O.
PSP6					I/O		Parallel Slave Port data.
RD7/PSP7	30	33	5	5		ST/TTL(3)	
RD7					I/O		Digital I/O.
PSP7					I/O		Parallel Slave Port data.
							PORTE is a bidirectional I/O port.
RE0/$\overline{RD}$/AN5	8	9	25	25		ST/TTL(3)	
RE0					I/O		Digital I/O.
$\overline{RD}$					I		Read control for Parallel Slave Port.
AN5					I		Analog input 5.
RE1/$\overline{WR}$/AN6	9	10	26	26		ST/TTL(3)	
RE1					I/O		Digital I/O.
$\overline{WR}$					I		Write control for Parallel Slave Port.
AN6					I		Analog input 6.
RE2/$\overline{CS}$/AN7	10	11	27	27		ST/TTL(3)	
RE2					I/O		Digital I/O.
$\overline{CS}$					I		Chip select control for Parallel Slave Port.
AN7					I		Analog input 7.
V_{SS}	12, 31	13, 34	6, 29	6, 30, 31	P	—	Ground reference for logic and I/O pins.
V_{DD}	11, 32	12, 35	7, 28	7, 8, 28, 29	P	—	Positive supply for logic and I/O pins.
NC	—	1, 17, 28, 40	12,13, 33, 34	13	—	—	These pins are not internally connected. These pins should be left unconnected.

Legend: I = input O = output I/O = input/output P = power
— = Not used TTL = TTL input ST = Schmitt Trigger input

Note 1: This buffer is a Schmitt Trigger input when configured as the external interrupt.
2: This buffer is a Schmitt Trigger input when used in Serial Programming mode.
3: This buffer is a Schmitt Trigger input when configured in RC Oscillator mode and a CMOS input otherwise.

SIM900
GSM/GPRS Module



The SIM900 is a complete Quad-band GSM/GPRS solution in a SMT module which can be embedded in the customer applications.

Featuring an industry-standard interface, the SIM900 delivers GSM/GPRS 850/900/1800/1900MHz performance for voice, SMS, Data, and Fax in a small form factor and with low power consumption. With a tiny configuration of 24mm x 24mm x 3 mm, SIM900 can fit almost all the space requirements in your M2M application, especially for slim and compact demand of design.

- SIM900 is designed with a very powerful single-chip processor integrating ARM926EJ-S core
- Quad - band GSM/GPRS module with a size of 24mmx24mmx3mm
- SMT type suit for customer application
- An embedded Powerful TCP/IP protocol stack.
- Based upon mature and field-proven platform backed up by our support service, from definition to design and production

General features

- Quad-Band 850/ 900/ 1800/ 1900 MHz
- GPRS multi-slot class 10/8
- GPRS mobile station class B
- Compliant to GSM phase 2/2+
  - Class 4 (2 W @ 850/ 900 MHz)
  - Class 1 (1 W @ 1800/1900MHz)
- Dimensions: 24* 24 * 3 mm
- Weight: 3.4g
- Control via AT commands (GSM 07.07, 07.05 and SIMCOM enhanced AT Commands)
- GSM application toolkit
- Supply voltage range 3.4 ... 4.5 V
- Low power consumption
- Operation temperature: -30 °C to +80 °C

Specifications for fax

- Group 3, class 1

Specifications for data

- GPRS class 10: max. 85.6 kbps (downlink)
- PBCCH support
- Coding schemes CS 1, 2, 3, 4
- CSD up to 14.4 kbps
- USSD
- Non-transparent mode
- PPP-stack

Specifications for SMS via GSM / GPRS

- Point-to-point MO and MT
- SMS cell broadcast
- Text and PDU mode

Drivers

- MUX Driver

Specifications for voice

- Tricodec:
  - Half rate (HR)
  - Full rate (FR)
  - Enhanced Full rate (EFR)
- Hands-free operation (Echo suppression)
- AMR
  - Half Rate(HR)
  - Full Rate(FR)

Interfaces

- Interface to external SIM 3V/ 1.8V
- analog audio interface
- RTC backup
- SPI interface
- Serial interface
- Antenna pad
- I2C
- GPIO
- PWM
- ADC

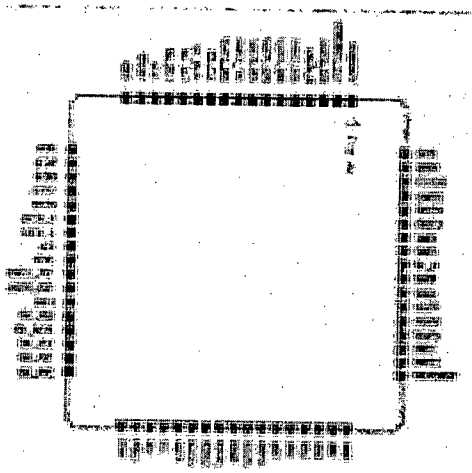
Compatibility

- AT cellular command interface

Approvals (in planning)

- CE
- FCC
- ROHS
- PTCRB
- GCF
- AT&T
- IC
- TA

Pin Assignment



More about SIM900 module, Please contact: Tel: +86 21 3252 3300
Fax: +86 21 3252 3301
E-mail: sim.com@sim.com

PROGRAMACION DE MICROCONTROLADOR

```
'*****************************************************************************

define OSC 4

'DEFINICIONES PARA LCD

Define  LCD_DREG       PORTD

Define  LCD_DBIT       4

Define  LCD_RSREG      PORTD

Define  LCD_RSBIT      3

Define  LCD_EREG       PORTD

Define  LCD_EBIT       2

'*****************************************************************************

ADCON1 = %00000110      ' CONFIGURAR PORTA COMO I/O DIGITALES   '
CONFIGURAR PORTA COMO I/O DIGITALES

TRISA = %11111100

TRISE.0=1:TRISE.1=1:TRISE.2=1

VALOR VAR BYTE

SALIDA VAR BYTE

'*****************************************************************************

'SALIDAS TIPO RELAY

SIRENA VAR PORTB.4:ENERGIZAR VAR PORTB.5:FOCO_ROJO VAR PORTA.0

FOCO_VERDE VAR PORTA.1:TRISB.1=1:TRISB.4=0:TRISB.5=0:TRISC=255

TRISB.6=0:TRISB.7=1

'*****************************************************************************

'PUERTOS PARA COMUNICACION CON MODEM GSM

TX VAR PORTB.6:RX VAR PORTB.7

'*****************************************************************************

'VARIABLES PARA COMANDOS O NUMEROS DE CELULAR A PROGRAMAR

D0 VAR BYTE:D1 VAR BYTE:D2 VAR BYTE:D3 VAR BYTE:D4 VAR BYTE:D5 VAR
BYTE
```

```
D6 VAR BYTE:D7 VAR BYTE:D8 VAR BYTE:D9 VAR BYTE:D10 VAR BYTE:TEMP1
VAR word

TEMP2 VAR word:TEMP3 VAR word:TEMP4 VAR word:clave var word:clave1  var
word

modem   VAR BYTE [8]:DATOS   VAR BYTE [11]:numero1 var byte[9]

numero2 var byte[9]:numero3 var byte[9]:numero4 var byte[9]

'*****************************************************************************

'VARIABLES AUXILIARES DE PROGRAMA

A VAR BYTE:SIR VAR BIT:ARMAR VAR BIT:CONTA VAR BIT:TEMPO VAR BIT:TP
VAR BIT

FV VAR BIT:POWER VAR BIT:CNT VAR BIT:A=4:CT1 VAR BIT:CT2 VAR BIT:CT3
VAR BIT

CT4 VAR BIT

'*****************************************************************************

LOW FOCO_ROJO:HIGH FOCO_VERDE:HIGH SIRENA :HIGH ENERGIZAR

'*****************************************************************************

tip var WORD:SIG VAR BYTE:REF VAR BYTE

CLAVE1=1234

'*****************************************************************************

'clave

write 36,1:write 37,2:write 38,3:write 39,4

read 50,A:READ 51,SIG:READ 52,REF:READ 54,FV:READ 90,CNT

READ 61,CT1:READ 62,CT2:READ 63,CT3:READ 64,CT4:

'*****************************************************************************

IF FV=1 THEN

   FOCO_VERDE=0

ELSE

   FOCO_VERDE=1

ENDIF

IF CT1=1 THEN
```

```
    LOW SIRENA
  ELSE
    HIGH SIRENA
  ENDIF
  IF CT2=1 THEN
    LOW ENERGIZAR
  ELSE
    HIGH ENERGIZAR
  ENDIF

  tip=0
  LCDOut $fe, 1
  LCDOut "  SEGURO MAX"
  PAUSE 2000
  '*****************************************************************************
  'PROGRAMAR MODEM
  LCDOut $fe, 1
  LCDOut "PROGRAM. MODEM"
  Pause 1000

 PROG_MODEM:
    SerOut2 TX,84,["AT+IPR=9600",13]
    PAUSE 2000
    SerOut2 TX,84,["AT+IFC=0,0",13]
    SerIn2 RX,84,2000,PROG_MODEM,[WAIT ("OK")]
    SerOut2 TX,84,["AT+CNMI=1,2,0,0,0",13]
    SerIn2 RX,84,2000,PROG_MODEM,[WAIT ("OK")]
    SerOut2 TX,84,["AT+CMGF=1",13]
    SerIn2 RX,84,2000,PROG_MODEM,[WAIT ("OK")]
```

```
SerOut2 TX,84,["AT+DDET=1",13]
SerIn2 RX,84,2000,PROG_MODEM,[WAIT ("OK")]
SerOut2 TX,84,["ATE0",13]
SerIn2 RX,84,2000,PROG_MODEM,[WAIT ("OK")]
SerOut2 TX,84,["ATS0=2",13]    'RESPUESTA AUTOMATICA DE VOZ DEL
MODEM  AL 2 RING
SerIn2 RX,84,2000,PROG_MODEM,[WAIT ("OK")]
SerOut2 TX,84,["AT&W",13]
SerIn2 RX,84,2000,PROG_MODEM,[WAIT ("OK")]
LCDOut $fe, 1
LCDOut "MODEM PROGRAMADO"
Pause 1000
'*****************************************************************************
REGISTRAR:
LCDOut $fe, 1
LCDOut  "REGISTRANDO..."
SerOut2 TX,84,["AT+CGREG?",13] ' PREGUNTA SI MODEN ESTA REGISTRADO
CON ALGUNA OPERADORA
SerIn2 RX,84,1000,REGISTRAR,[WAIT ("+CGREG:"),STR modem \4]
Pause 100
IF MODEM[3]="1" Then
   LCDOut $fe, 1
   LCDOut "MODEM REGISTRADO"
   Pause 1000
   Else
   GoTo REGISTRAR
  EndIF
  '*****************************************************************************
  GOSUB LEER_NUM
```

```
GOSUB VER_NUM

Inicio:

  SerIn2 RX,84,7000,ALARMA,[SKIP 10, wait("+"),STR DATOS \11\10]  'ESPERA
POR COMANDOS

  D0=DATOS[0]

  D1=DATOS[1]

  D2=DATOS[2]

  D3=DATOS[3]

  D4=DATOS[4]

  D5=DATOS[5]

  D6=DATOS[6]

  D7=DATOS[7]

  D8=DATOS[8]

  D9=DATOS[9]

  D10=DATOS[10]

  LCDOut $fe, 1

  LCDOut "Msg. Recibido!!!"

  LCDOut $fe,$C0,"DATO: ",D0, D1,D2,D3,D4,D5,D6,D7,D8,D9,D10

  PAUSE 200

'*****************************************************************************

   'SELECCION DE CASOS

   SELECT CASE D0

'*****************************************************************************

      case "D" ' CUANDO RECIBE COMANDO +DTMF:1 Ó +DTMF:2

        if d0="D" then

          IF D5="1" then

            LOW SIRENA

            CT1=1

            WRITE 61,CT1
```

```
endif
IF D5="2" then
   LOW ENERGIZAR
   CT2=1
   WRITE 62,CT2
endif

 IF D5="4" then
   HIGH SIRENA
   CT1=0
   WRITE 61,CT1
 endif
 IF D5="5" then
    HIGH ENERGIZAR
    CT2=0
    WRITE 62,CT2
  endif
  IF D5="3" then
     GoSub LEER_SW     ' RUTINA LEER ESTADO DE SW DE ARMAR Y
NIVEL
     GOSUB ENVIA_DATOS  'RUTINA QUE ENVIA DATOS POR SMS
   endif
   IF D5="6" then
     A=5:tip=0:FOCO_VERDE=0:write 50,5:WRITE 54,1
   endif
   IF D5="8" then
      A=4:tip=0:FOCO_VERDE=1:write 50,4:WRITE 54,0
   endif
   IF D5="9" then
```

```
            GOSUB LEER_NUM

            serout2 TX,84,["at+cmgs=",34,dec numero1[0],dec numero1[1],dec
numero1[2],dec numero1[3],dec numero1[4],dec numero1[5],dec numero1[6],dec
numero1[7],dec numero1[8],34,13]

             pause 1000

             serout2 TX,84,["NUMEROS DE CELULARES:",13,DEC
numero1[0],dec numero1[1],dec numero1[2],dec numero1[3],dec numero1[4],dec
numero1[5],dec numero1[6],dec numero1[7],dec numero1[8],13," ",DEC
numero2[0],dec numero2[1],dec numero2[2],dec numero2[3],dec numero2[4],dec
numero2[5],dec numero2[6],dec numero2[7],dec numero2[8],13," ",DEC
numero3[0],dec numero3[1],dec numero3[2],dec numero3[3],dec numero3[4],dec
numero3[5],dec numero3[6],dec numero3[7],dec numero3[8],26,13] 'ENVIO DE
MENSAJE

              PAUSE 1000

            endif

        ENDIF
'*****************************************************************************
  'PROGRAMAS NUMERO DE CELULAR QUE RECIBIRAN ESTADOS DE
ALARMA

      CASE "T"   'RTd1d2d3d4d5d6d7d8d9d10 ejemp: +T#968887721   comando
=(+T1951551591)

             numero4[0]=d2-48:numero4[1]=d3-48:numero4[2]=d4-48

             numero4[3]=d5-48:numero4[4]=d6-48:numero4[5]=d7-48

             numero4[6]=d8-48:numero4[7]=d9-48:numero4[8]=d10-48

           Select Case D1

             Case "1" ' se programa telefono 1

               write 100,numero4[0]

               write 101,numero4[1]

               write 102,numero4[2]

               write 103,numero4[3]

               write 104,numero4[4]

               write 105,numero4[5]

               write 106,numero4[6]
```

```
        write 107,numero4[7]
        write 108,numero4[8]
      Case "2" ' se programa telefono 2
        write 109,numero4[0]
        write 110,numero4[1]
        write 111,numero4[2]
        write 112,numero4[3]
        write 113,numero4[4]
        write 114,numero4[5]
        write 115,numero4[6]
        write 116,numero4[7]
        write 117,numero4[8]
      Case "3" ' se programa telefono 3
        write 118,numero4[0]
        write 119,numero4[1]
        write 120,numero4[2]
        write 121,numero4[3]
        write 122,numero4[4]
        write 123,numero4[5]
        write 124,numero4[6]
        write 125,numero4[7]
        write 126,numero4[8]
      End Select
      GOSUB LEER_NUM
      GOSUB VER_NUM
'*************************************************************************
    CASE "V"
      TEMP1=d1-48
      TEMP1=TEMP1*1000
```

```
      TEMP2=d2-48
      TEMP2=TEMP2*100
      TEMP3=d3-48
      TEMP3=TEMP3*10
      TEMP4=d4-48
      CLAVE1=TEMP1+TEMP2+TEMP3+TEMP4   'SE OBTIENE CLAVE

      if clave=clave1 then   ' SI LA CLAVE ES LA CORRECTA
        GoSub LEER_SW     ' RUTINA LEER ESTADO DE SW DE ARMAR Y
NIVEL
        GOSUB ENVIA_DATOS  'RUTINA QUE ENVIA DATOS POR SMS
        LCDOut $fe, 1     ' RUTINA PARA  INDICAR QUE SE ENVIO SMS
        LCDOut "Msg. Enviado!!!"
        PAUSE 1000

       endif

'*****************************************************************************

'LEE LOS CELULARES PROGRAMADOS
     CASE "G"   'COMANDO ES +G1234123456
       GOSUB LEER_NUM
       serout2 TX,84,["at+cmgs=",34,dec numero1[0],dec numero1[1],dec
numero1[2],dec numero1[3],dec numero1[4],dec numero1[5],dec numero1[6],dec
numero1[7],dec numero1[8],34,13]
       pause 1000
       serout2 TX,84,["NUMEROS DE CELULARES:",13,DEC numero1[0],dec
numero1[1],dec numero1[2],dec numero1[3],dec numero1[4],dec numero1[5],dec
numero1[6],dec numero1[7],dec numero1[8],13," ",DEC numero2[0],dec
numero2[1],dec numero2[2],dec numero2[3],dec numero2[4],dec numero2[5],dec
numero2[6],dec numero2[7],dec numero2[8],13," ",DEC numero3[0],dec
numero3[1],dec numero3[2],dec numero3[3],dec numero3[4],dec numero3[5],dec
numero3[6],dec numero3[7],dec numero3[8],26,13] 'ENVIO DE MENSAJE
```

```
      PAUSE 1000
'*************************************************************************
    CASE "A" 'ARMAR ALARMA PARA ENVIO DE 1 SMS DE ALERTA , comando
es: '+AccccXXXXXX
       TEMP1=d1-48:TEMP1=TEMP1*1000:TEMP2=d2-
48:TEMP2=TEMP2*100:TEMP3=d3-48
       TEMP3=TEMP3*10:TEMP4=d4-
48:CLAVE1=TEMP1+TEMP2+TEMP3+TEMP4
       if clave=clave1 then
         A=0:FOCO_VERDE=0:write 50,0:WRITE 54,1
       ENDIF
    CASE "B" 'ARMAR ALARMA PARA ENVIO DE 3 SMS DE ALERTA , comando
es: '+BccccXXXXXX
       TEMP1=d1-48:TEMP1=TEMP1*1000:TEMP2=d2-
48:TEMP2=TEMP2*100:TEMP3=d3-48
       TEMP3=TEMP3*10:TEMP4=d4-
48:CLAVE1=TEMP1+TEMP2+TEMP3+TEMP4
       if clave=clave1 then
         A=3:tip=0:FOCO_VERDE=0:write 50,3:WRITE 54,1
       ENDIF
    CASE "E" 'ARMAR ALARMA PARA ENVIO DE SMS CADA CIERTO TIEMPO
DE ALERTA , comando es: '+EccccXXXXXX
       TEMP1=d1-48:TEMP1=TEMP1*1000:TEMP2=d2-
48:TEMP2=TEMP2*100:TEMP3=d3-48
       TEMP3=TEMP3*10:TEMP4=d4-
48:CLAVE1=TEMP1+TEMP2+TEMP3+TEMP4
       if clave=clave1 then
         A=5:tip=0:FOCO_VERDE=0:write 50,5:WRITE 54,1
       ENDIF
    CASE "C" 'ARMAR ALARMA PARA NO ENVIO DE SMS DE ALERTA ,
comando es: '+CccccXXXXXX
       TEMP1=d1-48:TEMP1=TEMP1*1000:TEMP2=d2-
48:TEMP2=TEMP2*100:TEMP3=d3-48
```

```
        TEMP3=TEMP3*10:TEMP4=d4-
48:CLAVE1=TEMP1+TEMP2+TEMP3+TEMP4
        if clave=clave1 then
          A=4:tip=0:FOCO_VERDE=1:write 50,4:WRITE 54,0
        ENDIF
      CASE "W"
        TEMP1=d3-48:TEMP1=TEMP1*1000:TEMP2=d4-
48:TEMP2=TEMP2*100:TEMP3=d5-48
        TEMP3=TEMP3*10:TEMP4=d6-
48:CLAVE1=TEMP1+TEMP2+TEMP3+TEMP4
        if clave=clave1 then
          GoSub escribir_dig '+WHNccccXXXX,+WLNccccXXXX
        endif
    end select
  GoTo Inicio
  '*************************************************************************
  escribir_dig:
    Select Case D1
      Case "H"   '+WH1ccccXXX COMANDO
        SELECT CASE D2
          CASE "1"
            LOW SIRENA
            CT1=1
            WRITE 61,CT1
          CASE "2"
            LOW ENERGIZAR
            CT2=1
            WRITE 62,CT2
          END SELECT
```

```
Case "L"   '+WL1XXXXXXX COMANDO
   SELECT CASE D2
      CASE "1"
         HIGH SIRENA
         CT1=0
         WRITE 61,CT1
      CASE "2"
         HIGH ENERGIZAR
         CT2=0
         WRITE 62,CT2

   END SELECT
End Select
Return
'*****************************************************************************

ENVIA_DATOS:
   'ENVIA DATOS SOLICITADOS CON COMANDO +V1234123456
   serout2 TX,84,["at+cmgs=",34,dec numero1[0],dec numero1[1],dec
numero1[2],dec numero1[3],dec numero1[4],dec numero1[5],dec numero1[6],dec
numero1[7],dec numero1[8],34,13]
   pause 1000
   serout2 TX,84,["ARMAR: ",DEC1 ARMAR," VA:",DEC1 A,"E:", DEC1
VALOR.0,DEC1 VALOR.1,DEC1 VALOR.2,DEC1 VALOR.3,DEC1 VALOR.4,DEC1
VALOR.5,DEC1 VALOR.6,DEC1 VALOR.7," S:", DEC1 SALIDA.0,DEC1
SALIDA.1,26,13] 'ENVIO DE MENSAJE
   PAUSE 5000
   return
'*****************************************************************************

ALARMA:
   GOSUB LEER_SW  ' LEER SW QUE HABILITA O ARMA LA ALARMA
   LCDOUT $FE,1
```

```
  LCDOUT "E:", DEC1 VALOR.0,DEC1 VALOR.1,DEC1 VALOR.2,DEC1
VALOR.3,DEC1 VALOR.4,DEC1 VALOR.5,DEC1 VALOR.6,DEC1 VALOR.7," S:",
DEC1 SALIDA.0,DEC1 SALIDA.1

  LCDOUT $fe,$C0, "A:",DEC1 ARMAR," VA:",DEC1 A

  PAUSE 100

  IF ARMAR=1 THEN

    if VALOR <> 0 then

      LOW SIRENA' ACTIVA SIRENA

      GOSUB MENSAJE_AVISO  ' ENVIA MENSAJE DE ALERTA A 3
NUMMEROS CONFIGURADOS

    ELSE

      high SIRENA ' DESACTIVA SIRENA

    ENDIF

  ENDIF


GOTO INICIO
'*****************************************************************************

LEER_SW:

  'IF PORTB.1=0 and FV=1 THEN

   '  FOCO_VERDE=1

    ' FV=0

     ' A=4

      ' tip=0

      ' write 50,4

      ' WRITE 54,FV

      ' PAUSE 2000

    'ENDIF

    'IF PORTB.1=0 and FV=0 THEN

     '  FOCO_VERDE=0

      ' FV=1
```

```
' A=5
'tip=0
' write 50,5
' WRITE 54,FV
' PAUSE 2000
'ENDIF
ARMAR=~FOCO_VERDE
VALOR.0=~PORTC.0
VALOR.1=~PORTC.1
VALOR.2=~PORTC.2
VALOR.3=~PORTC.3
VALOR.4=~PORTC.4
VALOR.5=~PORTC.5
VALOR.6=~PORTC.6
VALOR.7=~PORTC.7
SALIDA.0=~PORTB.4
SALIDA.1=~PORTB.5
Return
'*****************************************************************************
MENSAJE_AVISO:
'*****************************************************************************
    IF A=0 THEN
      gosub Anumero1
      GOSUB LEER
      gosub Anumero2
      GOSUB LEER
      gosub Anumero3
      GOSUB LEER
      A=4
```

```
    write 50,4
    FOCO_VERDE=1
    FV=0
    WRITE 54,FV
  ENDIF
'*************************************************************************
  IF A=3 THEN
    tip=tip+1
     pause 100
     select case tip
     case 10
        gosub envio_sms
        case 200
        gosub envio_sms
        case 400
        gosub envio_sms
        tip=0
        a=4
        write 50,4
        FOCO_VERDE=1
        FV=0
        WRITE 54,FV
       end select
     ENDIF
   '*************************************************************************
      IF A=5 THEN
        tip=tip+1
        pause 100
        select case tip
```

```
      case 2
      gosub envio_sms
      case 200
      gosub envio_sms
      tip=3
    end select
  ENDIF
RETURN
envio_sms:
      gosub Anumero1
      GOSUB LEER
      gosub Anumero2
      GOSUB LEER
      gosub Anumero3
      GOSUB LEER
 return

 Anumero1:
   serout2 TX,84,["at+cmgs=",34,dec numero1[0],dec numero1[1],dec
numero1[2],dec numero1[3],dec numero1[4],dec numero1[5],dec numero1[6],dec
numero1[7],dec numero1[8],34,13]
   pause 1000
   serout2 TX,84,["ALARMA ACTIVA 1!!! ",10,13,"E:", DEC1 VALOR.0,DEC1
VALOR.1,DEC1 VALOR.2,DEC1 VALOR.3,DEC1 VALOR.4,DEC1 VALOR.5,DEC1
VALOR.6,DEC1 VALOR.7," S:", DEC1 SALIDA.0,DEC1 SALIDA.1,26,13] 'ENVIO DE
MENSAJE
   return

  Anumero2:
```

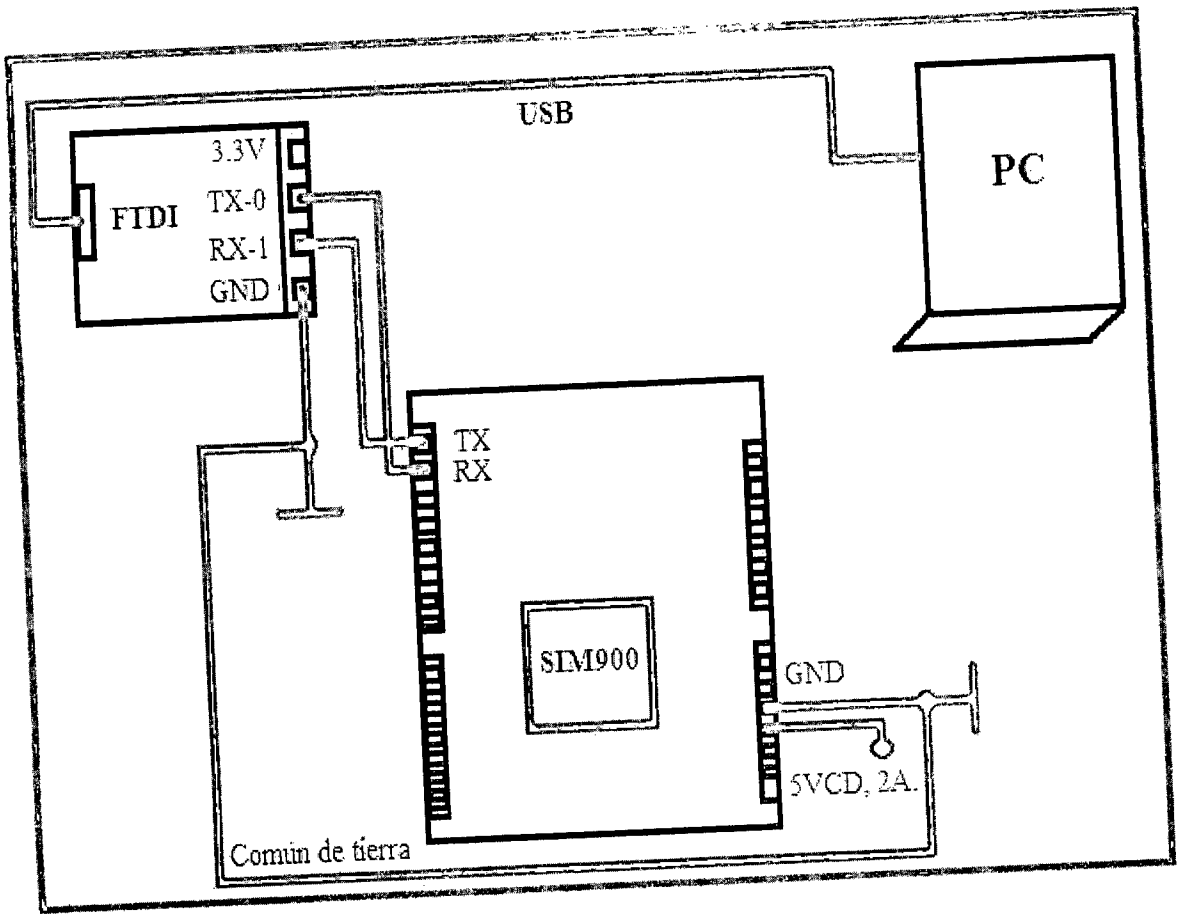
```
  serout2 TX,84,["at+cmgs=",34,dec numero2[0],dec numero2[1],dec
numero2[2],dec numero2[3],dec numero2[4],dec numero2[5],dec numero2[6],dec
numero2[7],dec numero2[8],34,13]

  pause 1000

  serout2 TX,84,["ALARMA ACTIVA 1!!! ",10,13,"E:", DEC1 VALOR.0,DEC1
VALOR.1,DEC1 VALOR.2,DEC1 VALOR.3,DEC1 VALOR.4,DEC1 VALOR.5,DEC1
VALOR.6,DEC1 VALOR.7," S:", DEC1 SALIDA.0,DEC1 SALIDA.1,26,13] 'ENVIO DE
MENSAJE

  return


Anumero3:

  serout2 TX,84,["at+cmgs=",34,dec numero3[0],dec numero3[1],dec
numero3[2],dec numero3[3],dec numero3[4],dec numero3[5],dec numero3[6],dec
numero3[7],dec numero3[8],34,13]

  pause 1000

  serout2 TX,84,["ALARMA ACTIVA 1!!! ",10,13,"E:", DEC1 VALOR.0,DEC1
VALOR.1,DEC1 VALOR.2,DEC1 VALOR.3,DEC1 VALOR.4,DEC1 VALOR.5,DEC1
VALOR.6,DEC1 VALOR.7," S:", DEC1 SALIDA.0,DEC1 SALIDA.1,26,13] 'ENVIO DE
MENSAJE

  return
'*****************************************************************************

'LEER NUMERO DE CELULARES Y CLAVE

LEER_NUM:
   read 100,numero1[0]
   read 101,numero1[1]
   read 102,numero1[2]
   read 103,numero1[3]
   read 104,numero1[4]
   read 105,numero1[5]
   read 106,numero1[6]
   read 107,numero1[7]
   read 108,numero1[8]
```

```
read 109,numero2[0]
read 110,numero2[1]
read 111,numero2[2]
read 112,numero2[3]
read 113,numero2[4]
read 114,numero2[5]
read 115,numero2[6]
read 116,numero2[7]
read 117,numero2[8]

read 118,numero3[0]
read 119,numero3[1]
read 120,numero3[2]
read 121,numero3[3]
read 122,numero3[4]
read 123,numero3[5]
read 124,numero3[6]
read 125,numero3[7]
read 126,numero3[8]

read 36,modem[4]
read 37,modem[5]
read 38,modem[6]
read 39,modem[7]

'*************************************************************************

'CLAVE
TEMP1=MODEM[4]
TEMP1=TEMP1*1000
```

```
 TEMP2=MODEM[5]

 TEMP2=TEMP2*100

 TEMP3=MODEM[6]

 TEMP3=TEMP3*10

 TEMP4=MODEM[7]

 CLAVE=TEMP1+TEMP2+TEMP3+TEMP4  'SE OBTIENE CLAVE PARA ARMAR
ALARMA

RETURN

VER_NUM:

  LCDOut $fe, 1

  LCDOut " Te1:",dec numero1[0],dec numero1[1],dec numero1[2],dec
numero1[3],dec numero1[4],dec numero1[5],dec numero1[6],dec numero1[7],dec
numero1[8]

   Pause 1000

   LCDOut $fe, 1

   LCDOut " Te2:",dec numero2[0],dec numero2[1],dec numero2[2],dec
numero2[3],dec numero2[4],dec numero2[5],dec numero2[6],dec numero2[7],dec
numero2[8]

    Pause 1000

    LCDOut $fe, 1

    LCDOut " Te3:",dec numero3[0],dec numero3[1],dec numero3[2],dec
numero3[3],dec numero3[4],dec numero3[5],dec numero3[6],dec numero3[7],dec
numero3[8]

     Pause 1000

     LCDOut $fe, 1

     LCDOut " LA CLAVE ES:"

     LCDOUT $FE,$C0,dec4 clave

     PAUSE 1000

     LCDOut $fe, 1

   RETURN

   '*****************************************************************************
```

```
LEER:
 FOR D8=0 TO 10
   GOSUB LEER_SW
   PAUSE 200
 NEXT D8
 RETURN
 '*****************************************************************************
```

COMANDOS AT


USB
PC
FTDI
3.3V
TX-0
RX-1
GND
TX
RX
SIM900
GND
5VCD, 2A.
Comun de tierra

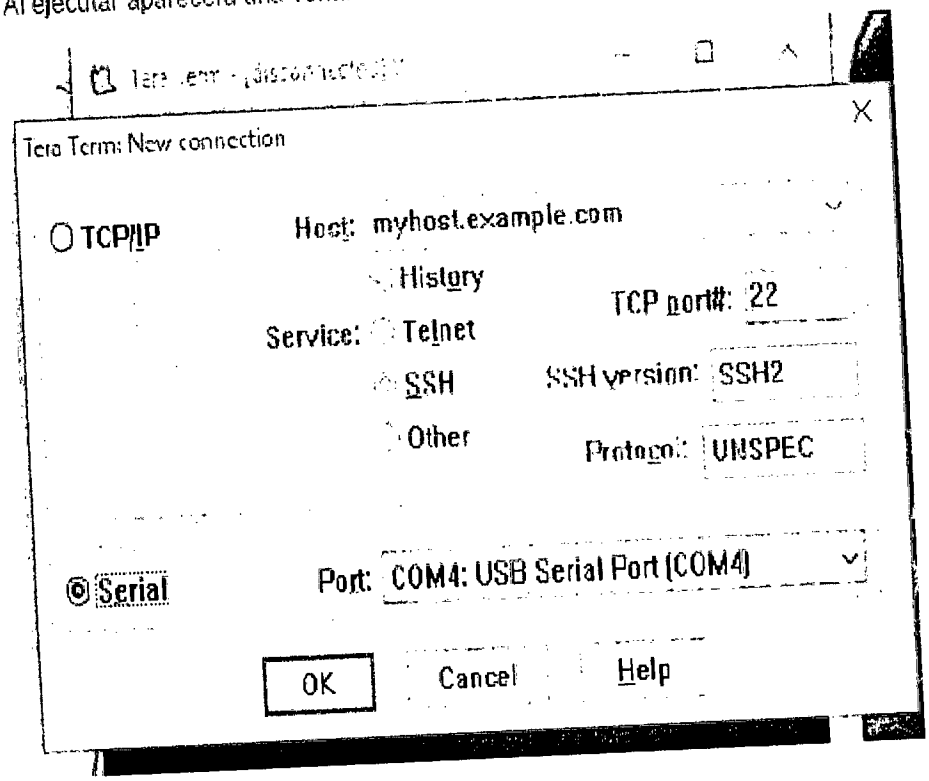
Al conectarnos, es importante verificar

PC → Panel de control → Administrador de Dispositivos → Puertos COM y LTP.

El puerto COM en el que estará operando el FTDI una vez instalado.

Abrimos una terminal de RS232, se recomienda Tera Term VT.

Al ejecutar aparecerá una ventana



1. Seleccionar Serial → Port: COM
   En este caso se pone el COM en el que haya caído el FTDI.

2. Pulsa en la barra de herramientas Set Up →Terminal...
   Selecciona CR+LF y pulsa OK.

Una vez conectados y si el módulo SIM900 está alimentado correctamente, el primer LED aparecerá en ROJO así procederemos a energizar la tarjeta

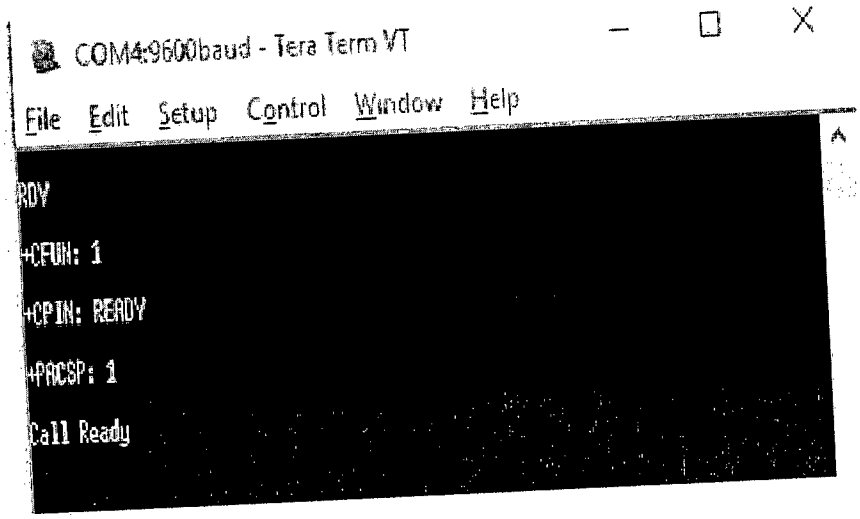
- presionemos al menos 2 segundos el primer botón que dice "SIM900-POWER".

El segundo LED se encenderá en color *VERDE*

El tercero permanecerá pulsando constantemente.

Al SIM900 se le añade un chip de teléfono de cualquier compañía celular.

Si todo está correcto, en la interfaz Tera Term aparecerá de la siguiente manera la comunicación de nuestro modulo SIM



¡Felicidades!

Estás listo para empezar a enviarle comandos AT al SIM900

¡Comsat

Recuerda, al mandar un comando AT presionar tecla *Enter* para mandar una orden, excepto al envío de un mensaje.

Comandos

AT
Sirve para verificar si el módulo SIM900 está funcionando adecuadamente para entrar en modo comando.
Al enviar AT el SIM deberá contestarnos con un OK.

AT+CGMI
Veremos en nombre del fabricante

ATI
Ver la información del producto.

AT+IPR=?
Preguntar el Baud Rate en el que puede operar el SIM

AT+IPR?
Sirve para preguntar el Baud Rate actual

AT+IPR=XXXX
Configuremos a la frecuencia deseada

AT+COPS?
Nombre de la compañía telefónica

AT+CGSN
Visualizar el IMEI del chip utilizado

AT+CSCS?
Tipo de texto

AT+CSCS="XXX"
Configurar a tipo de texto

AT+CMGF?
Ver el formato de un mensaje. ya sea PDU(0) o SMS(1)"

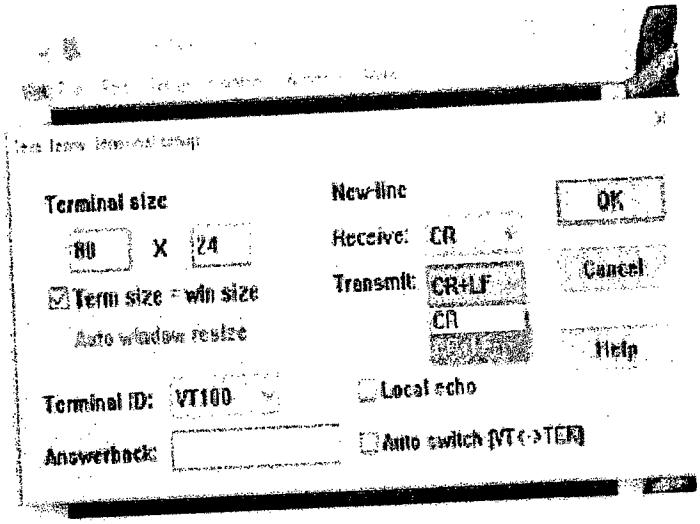
AT+CMGS=04455XXXXXXX
Enviar un SMS Se despliega el símbolo mayor que > Escribir mensaje y al finalizar presiona Ctrl+Z retornará OK si el SMS se envió correctamente.

AT+CMGL=ALL
Sirve para ver todos los mensajes que nos han llegado al SIM

ATD04455XXXXXXX;
Sirve para hacer una llamada a cualquier teléfono móvil

ATA
Sirve para contestar una llamada

ATH
Sirve para colgar una llamada



Selecciona el Baud Rate en el que desea trabajar con el módulo SIM900. Se recomienda 9600-115200.

3. Set Up → Serial Port